Engineering Requirements for the Development of Cloud Based Support for Autonomous Trucks

A Design Science Study

Master's thesis in Computer Science and Engineering

SUPRIYA SUPRIYA

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2020

MASTER'S THESIS 2020

Engineering Requirements for the Development of Cloud Based Support for Autonomous Trucks

A Design Science Study

SUPRIYA SUPRIYA

UNIVERSITY OF GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2020

Engineering Requirements for the Development of
Cloud Based Support for Autonomous Trucks
A Design Science Study

SUPRIYA SUPRIYA

Supervisor: Eric Knauss, Department of Computer Science and Engineering
Advisor: Md Abdullah Al Mamun, Volvo Autonomous Solution
Examiner: Regina Hebig and Michel Chaudron, Department of Computer Science and Engineering

Master's Thesis 2020
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2020

Engineering Requirements for the Development of
Cloud Based Support for Autonomous Trucks
A Design Science Study

SUPRIYA SUPRIYA
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

[**Context and Motivation**] Continuous development is about making software development a continuous endeavour rather than a one time process, for the solution to be ready for release all the time and also to grow and evolve in small increments. With the recent advancements in the remote cloud based support a.k.a control tower for the autonomous trucks, automotive original equipment manufacturers, which traditionally struggle to embrace continuous development paradigms are embracing the phenomena from early stages in requirements to remain competitively ahead in business.
[**Question/Problem**] Despite many practices in requirements engineering there is lack of an effective requirements practice that could be applicable for autonomous vehicle support software. Especially, when considering continuous integration or deployment, it makes it hard to acquire good representative set of use cases which could be structured and maintained before dwelling into actual implementation. Along with it, there is lack of information on how the operational design domain elements of the vehicle could relate to the use cases.
[**Principal ideas/results**] Based on the design science research approach, this study as an artifact provides an expandable yet useful initial list of use cases to inform continuous development, a use case classification, list of challenges in the control tower implementation, an information model that relates these use cases to relevant other information (e.g, about operational design domain elements, quality attributes, etc.) and a novel template for the generation of effective use cases. The findings from evaluation show that the artifact is useful and applicable in the development.
[**Contribution**] The major contribution is the designed artifact itself, which contributes in the field of requirements engineering and is a solution to the industrial context verified from its practical applicability. The artifact not only enables novel solution to the identified problems but also contributes to the design science knowledge base.

Keywords: use cases, requirements engineering, control tower, cloud support, design science research, artifact, autonomous trucks, continuous development, use case template, information model, operational design domain.

Acknowledgements

This thesis has been performed within the Department of Computer Science and Engineering, for the division of Software Engineering, at Chalmers University of Technology. The work has involved close correspondence with industry representatives who have significantly conduced to the result.

I would like to thank my supervisor Eric Knauss for his immense support through constructive feedback and generously sharing his time. I thank him for involving me in research meetings and introducing to research connections. My sincere appreciation to industry advisor Abdullah Mamun for his feedback and engagement throughout the process. I would like to thank my examiner Regina Hebig for evaluating my final thesis and for her time and constructive feedback. Thanks to my previous examiner Michel Chaudron for being able to guide and engage during first half of the thesis. I would like to thank my friend Tobias Schwarz for peer reviewing my work and being part of the thesis defence. Finally, a sincere thank you to all industry participants who have shared their thoughts and expertise for the benefit of study conclusions.

Supriya Supriya, Gothenburg, September 2020

Contents

1 Introduction

The technology in automotive domain has far emerged from what it was few years back. Steps are being laid from traditional human driven into (semi-)autonomous drive where the vehicles have human-like capability. From collecting data, interpreting the information to committing actions, that would have been done similar by a human, these technologies are far emerging.

With the car manufactures being forefront regarding prior advancements in autonomous cars[1], the automotive industries have also witnessed need in autonomous trucks. The economic case for driver-less technology is way higher for commercial vehicles[2] as they could make the entire process efficient and increase productivity. But similar to autonomous cars, this also leads to the concern of safety and ethics during drive especially during boundary cases, where typically the human decision-maker would exercise some individual moral judgement (Burton et al., 2020).

From the perception of surrounding environment and predefined control, the autonomous trucks can take decisions in their operational design domain (ODD) environment. But when real situations dynamically interact, for example a bad weather situation, it becomes a necessity to have a control tower to monitor and coordinate the activities of the trucks by giving extra input to vehicles beyond what their sensors can perceive. On a larger futuristic scale, when providing dynamic fleet based autonomous solutions, it also becomes necessary to have a central unit to control fleet operation (Vagia et al., 2019). It is also a need to plan and optimize missions according to the dynamic transportation demands of the users and depending upon within the different periods of the day (Bsaybes et al., 2019).

With the development of communication technologies, cloud solutions and digitization, there is a shift from traditional servers to cloud based support, allowing the control towers to be remotely located, adding to the further improvement in efficiency. This opens unlimited possibility of data-collection from not just the sensors and actuators of autonomous trucks during commercial operation but also the control tower which can collect data from external sources, e.g., weather stations, traffic management authorities, other autonomous vehicle manufacturers, etc. These factors open wide possibilities to research on the wide variety of functionalities that these towers can provide to autonomous trucks using the collected data.

[1] You can now hail a ride in a fully autonomous vehicle, courtesy of Waymo (Accessed on 2020-07)

[2] Trucking and Logistics Will Lead the Autonomous Vehicle Revolution (Accessed on 2020-07)

Continuous software development (CD) has received greater attention in recent years (Bosch, 2014). The solution developed using CD makes the software "always" ready for release contrary of the agile way of making the software ready for release. This ambition of the company to adapt CD to remain forefront of its competitors, opens possibilities to explore requirements engineering practices in the area of CD.

The aim of this master thesis is to apply the knowledge gained in requirements engineering to support the continuous development of the control tower of autonomous trucks. The study involves identification of existing control tower use cases and the possible challenges for the trucks of Level 4 autonomy[3]. The study provides a template for structuring and documenting use cases effectively. Along with it, the study aims to discover the relation of the control tower use cases mainly with ODD, data requirements and quality attributes. The thesis thus introduces use cases in the CD practice and way to write them effectively. The research in the usage of use cases in CD and the research in the control tower area is still new and this thesis can be a contribution to the requirements engineering field and booming autonomous sector and help other researchers and practitioners in near future.

1.1 Statement of the Problem

Studies in requirements engineering in relation to CD, has its focus mainly with using user stories (Niu et al., 2018). Studies show good practices such as establishing a definition of done for each user story, linking user stories to requirements and tests, etc. (Kasauli et al., 2017). Despite numerous information sources in requirements engineering, we do not yet know a solution that could be applied for continuous development of autonomous vehicle support software. Since the area of this thesis is new with the very few researches out and with a possibility to capture large variety of specifications of autonomous driving, a use case can be good tool but yet to be explored. Along with it, since the control tower functionalities can be broader and unknown, a way to write use cases effectively in this expanding domain is a challenge.

An ODD imposes numerous restrictions on the autonomous vehicles (AVs) (Czarnecki, 2018a). Despite the limitation, the number of ODD components is high in number which makes finding the appropriate components challenging. The use cases are by themselves interesting. But also, a method to connect the use cases and the ODD's are interesting, desperately needed for autonomous drive.

1.2 Statement of Purpose

The purpose of the study is to provide input to the off-board control system a.k.a control tower of autonomous trucks, by identifying the most important use cases and classifying them based on various challenges. The use cases, its classification and the

[3]SAE Standards News: J3016 automated-driving graphic update (Accessed on 2020-07)

challenges in implementing them should be able to guide the implementation road map of the new software and services in relation to the continuous development of the control tower. This study aims to introduce use cases in the CD practices and provide a use case template. Further, use cases related to autonomous drive strongly relate to a specifically defined context, often expressed as ODD. It is however unclear how to best specify use cases together with ODDs and it is also the purpose of this thesis to uncover good practices for it.

1.3 Case Company

This thesis is carried out in collaboration with the company Volvo Autonomous Solutions, Sweden[4]. They have developed autonomous trucks to be used in mining areas and are interested to develop solutions to extend the operation of autonomous trucks into mixed traffic. The development of trucks in level 4 has arisen need for control tower development that would enhance the vehicles operational capabilities and increase safety on road. By engineering effective requirements practices and solutions, this study aims to support the continuous development of the control tower.

1.4 Research Questions

The study objective is designed to identify the problems on lack of effective use cases in the continuous development of the control tower, find solutions to it and evaluate the extent to which the study findings are able to solve the identified problems. This objective is divided into 3 research questions as in below which are answered and developed in iterations using design science research (DSR) methodology.

RQ1: *Which problems are encountered when specifying use cases to support continuous development of off-board control system of autonomous trucks?*

RQ2: *Which solutions could mitigate this problem?*

RQ3: *To what extent can the problems (RQ1) be solved by potential solutions (RQ2)?*

[4]Volvo Autonomous Solutions

2
Background and Related Work

This chapter aims to discuss the concepts and literature's in relation to the study. The chapter begins with the discussion of use cases which is the main concept around which the artifact is designed. It is followed with discussion on continuous development and its practices in requirements engineering. Finally the chapter ends with terms in relation to autonomous vehicles with description of control tower, levels of autonomy and operational design domain.

2.1 Use Cases

Cockburn (2001) defines use case as a means to capture system behaviour that serve as a means of communication among people. They should be simple to understand and provide flexibility to describe the system functionality through varying levels (cloud level, sea level, etc.).

For use cases specific to autonomous vehicles, ADAS&ME[1] lists 7 use cases. These use cases provide advanced driver assistance services to all vehicle types. But the use cases are functionalities that the autonomous vehicle will be self capable of executing as the level of autonomy increases. Similarly, the 4 use cases discussed by Wachenfeld et al. (2016) and the use cases for autonomous vehicles based on collision avoidance driver support functionality are specific for the vehicle in its on-board system. Rupp et al. (2010) also discusses advanced functionalities specific for AVs and not for control tower. However, few literature's (Whaiduzzaman et al. (2014), Stephan et al. (2011), Kumar et al. (2012), Milani et al. (2019)) mention the functionalities that could be provided to autonomous vehicles remotely or the features that could be shifted from the on-board of the vehicle to off-board. But existing works are not full fledged. Functionalities that were found relevant are extracted from the existing work and described in Chapter 4.

Use case template

Use case is merely a form of writing. But they come with different forms and structured in different types of templates (Cockburn, 2001). The templates vary from informal textual description to a more formal structured specification. A higher degree of formalism improves the completeness, the consistency, the correctness of the specification, which also decreases the redundancy in the specification (Tiwari et al.,

[1]Use cases - ADAS&ME (Accessed on 2020-07)

2017). Still, the specification of the use cases to be complete, correct and provide consistent set of requirements has remained a major issue (Anda et al., 2009).

In order to deliver value to customer in quicker pace during continuous development, documenting use cases should not be a problem. For this purpose, there is wide spread usage of user stories to capture requirements in structured way (Niu et al., 2018). As Ian Sommerville states, *"People love stories. People relate to stories"*. However, a use case documented in a template can provide a structured format Cockburn (2001). But to achieve a concise structured format, the elements in the template play a major role. Tiwari et al. (2017) in a literature review study lists the elements used in different use case templates by the authors as seen in Figure 2.1. These elements or properties are many in number and can make documentation a tedious task. This could be a reason for user stories to be popular as stories are interesting and get straight to the point which people can understand. There lies a void in usage of use case template in continuous development practice and in finding a template with only necessary elements to write an effective use case.

Use case Element/Properties	Cockburn	Tiwari	Insfrán	Jacobson	Somé	Yue	Kruchten	Kettenis
Use case name	✓	✓	✓	✓	✓	✓	✓	✓
Goal in context (Description)	✓	✓	✓	✓	✓	✓	✓	✓
Trigger	✓						✓	✓
Scope								✓
Summary	✓							
Level of abstraction	✓							✓
Stakeholders	✓							✓
Stakeholders and their interests	✓							✓
Primary actor	✓	✓	✓		✓	✓	✓	✓
Secondary actors		✓	✓		✓	✓	✓	
Special Requirement	✓		✓				✓	✓
Frequency of occurrence of use case	✓							
Priority	✓							
Technology and Data Variations	✓							
Precondition	✓	✓		✓	✓	✓	✓	✓
Postconditions (Success)	✓	✓		✓	✓	✓	✓	✓
Postconditions (Failure)	✓							✓
Super use case	✓							
Sub use case	✓							
Dependency		✓				✓		
Generalization		✓				✓		
Basic flow	✓	✓	✓	✓	✓	✓	✓	✓
Sub flow		✓		✓				
Alternate flow	✓	✓	✓	✓	✓	✓	✓	✓
Exceptional Flow	✓	✓						
Extension Points	✓	✓				✓	✓	✓
Variations	✓	✓						✓
Cross reference to high-level requirements in SRS			✓					
Open Issue(s)	✓							✓
Use of Keywords		✓			✓	✓	✓	
Postcondition for each alternate flow	✓					✓	✓	✓
Conditions for each flow of events to happen		✓						✓
Adjacency Pairs		✓	✓					
Control flow structures for iteration	✓	✓	✓		✓	✓	✓	✓
Information about Concurrent transactions		✓		✓	✓	✓		

Figure 2.1: Elements present in the use case templates. Adapted from Tiwari et al. (2017)

2.2 Continuous Development

Continuous Development or Continuous Delivery is built upon the previous agile methodologies such as DevOps, Continuous Integration and Agile. Here the software is continuously developed in iterations and is made to be available for release all the time.

There exists reasons that make CD different from agile practices. The agile methodology is about developing a product which needs to be finished before release. The end product must have all the intended functionalities. This is not the same with CD. Unfinished products can be released to the users but the unfinished product must have basic functionality to be users satisfaction.

The agile functions by breaking the significant problem into smaller chunks and resolving them individually. In CD, the technique known as feature toggle[2] enables developers and the entire team to meet the customers requirements even without finishing the entire software. The unfinished features are hidden so that they do not appear to the users. With this, the code remains deployed simultaneously as the work progresses by adding new features in each iteration. This makes it continuous and eliminates the need for frequent releases.

Companies adopt CD as it minimizes risk or loss by enabling them to release the software whenever they want. CD improves efficiency and enables to reach the market faster than the other competitors. Facebook has increased its mobile deployments from 8 weeks to deployment every week and the findings reveal that it increased the software quality as the developers were less likely to rush their software pushes as they knew that another release would follow shortly (Savor et al., 2016). In order to reach faster execution, technical aspects such as integration are automated. It automatically deploys every build that passes CI, which shortens the development cycle. All these practices are at the technical level.

From the requirements engineering perspective, Krusche et al. (2014) presents workflow that will help the developers to validate their understanding of the requirements. The workflow enables to deliver the software with fewer clicks in order to receive faster feedback from the customers. However there exists challenges that limit the companies in adapting CD. Shahin et al. (2017) reveals 11 confounding factors with the most important ones being lack of automated (user) acceptance test, manual quality check, deployment as business decision, insufficient level of automated test coverage and highly bureaucratic deployment process. However we do not know the problems that companies face during the initial stages of CD especially specific to control tower development nor what requirements engineering practices exist or can be improved to support CD.

The SAFe requirements model[3] includes continuous exploration as the initial part

[2]Feature toggle (Accessed on 2020-07)
[3]Scaled Agile framework (Accessed on 2020-07)

of the continuous delivery pipeline. It defines continuous exploration as the process that drives innovation and fosters alignment on what should be built by continually exploring market and customer needs, and defining a vision, road map, and set of features for a solution that addresses those needs. The definition quite aligns with the thinking and we look into how use cases can be used in this process.

2.3 Control Tower for Autonomous Vehicles

A remote control from a station or a tower has been used previously in the field of aviation, marine, railways, etc. In aviation, recent advancements suggest to move towards remote air traffic control from traditional physical equipment of control towers in airport. Cordeil et al. (2016) showed two immersive air traffic control/management prototypes for control and data data analysis which can work remotely. Luftfartsverket and SAAB have developed remote tower services with Örnsköldsvik Airport in Sweden being the first in the world to have this service[4]. Having a controller to monitor and control air traffic is seen to benefit safety and effective management of air traffic (Schmidt et al., 2009).

In marine, autonomous submersibles have been developed to be controlled remotely, without requiring an operator to be present in them. This helps to explore wide difficult range of operations such as deep-ocean and under-ice exploration, tasks in hazardous areas, in natural or man made disastrous regions, automated searches, surveillance missions, etc (Yuh et al., 2011). Furthermore, unmanned surface vehicles (USVs) also known as autonomous surface vehicles improved personnel safety and security, extended operational range and precision as well as increased flexibility in sophisticated environments. These are remotely controlled through ground stations located in an onshore facility, a mobile vehicle or an offshore ship, which monitor real time status of USV and its on board equipment (Z. Liu et al., 2016).

In railways a train traffic controller is used in the reduction of travel delays, improved timeliness and efficient use of infrastructure (Kauppi et al., 2017). Similar to an air traffic control tower that monitors flights, the autonomous vehicle control tower should be able to monitor and take action upon unexpected behaviour. However, the main difference between control tower for autonomous vehicles and other automated transports (aviation, marine, railway, etc.) is the intensive road network and complex infrastructure (Zhao et al., 2019).

Apart from this, the autonomous vehicles face challenge on road when complicated traffic rules and real situations in the environment interact dynamically. Example of such would be, temporary important signs placed on road and the weather condition is poor which makes the vehicle difficult to interpret the sign accurately in real time. This makes the operation of autonomous trucks challenging, especially when carrying cargo that must be delivered on time, over long distances across lands, etc. These complex dynamic situations raises need for additional support for the vehicle,

[4]Sweden first in the world with remotely operated air traffic management (Accessed on 2020-07)

which is the reason behind having a control tower.

An automated vehicle traffic control tower (AVTCT) for autonomous driving system (ADS) coordinates among fleet of AVs, infrastructures, service providers and traditional road users. It improves traffic efficiency and safety by assisting AVs with difficult situations[5]. However, it is necessary to test possible functionalities and benefits of AVTCT, as well as reveal challenges in applying it. Figure 2.2 shows the design of the idea of a control room for multiple teleoperators monitoring the vehicles through displays ahead.

Figure 2.2: Control tower room for automated vehicles, from ITRL et al. (2018)

2.4 Levels of Autonomy

The Society of Automotive Engineers (SAE) defines 6 levels of autonomy in vehicles ranging from Level 0 to Level 5[6] . Vehicles in Level 0 are manually controlled with no automation and Level 5 being fully autonomous. Figure 2.3 shows the levels ranging from 1 to 5 along with the features. Level 0 is not included in the figure because although there exists systems in the Level 0 vehicles to help the driver, they would technically wont drive the vehicle, thus not qualifying for automation. An example of such would be the emergency braking system.

For this study, the focus is on cloud based support for trucks operating in Level 4 autonomy. The main difference between Level 3 and Level 4 vehicles is that in Level

[5]AVTCT (Accessed on 2020-07)

[6]Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles (Accessed on 2020-07)

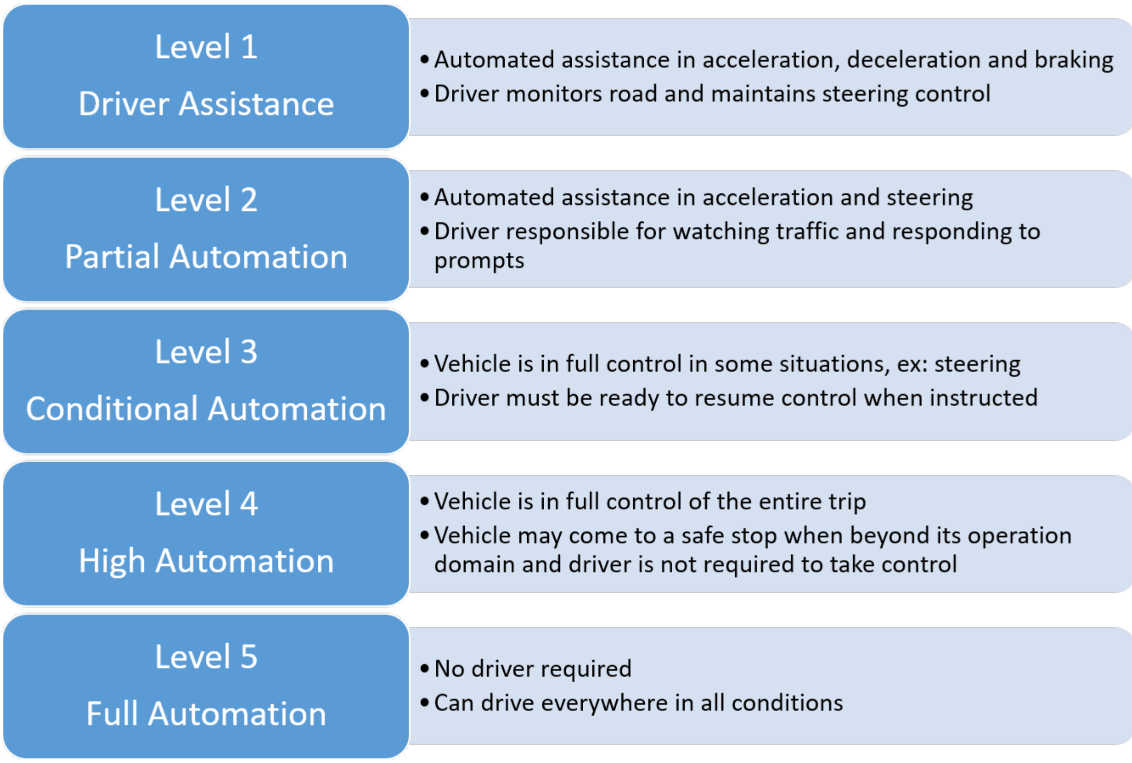

Figure 2.3: Five levels of autonomy in vehicles

4, vehicles can understand if there is any system failure or situations out of vehicles capability. The vehicles are programmed to understand their operation environment such as operating autonomously even during moderate rain but not during heavy snow. When the vehicle is not designed to operate in such adverse environment conditions, it is capable to sense it and come to a safe stop. The vehicle can either wait until it senses back its driving conditions or the driving could be resumed by a teleoperator in the control tower operating the vehicle remotely.

The need for control tower in Level 5 autonomy will be minimum, as the truck will be able to go anywhere and do anything that an experienced human driver can do. There shall not be any teleoperator required as the truck will require no external assistance and can operate in any external conditions.

2.5 Operational Design Domain

An Operational Design Domain is a term used to describe vehicles in Level 3, 4 or 5 autonomy. SAE J3016 defines ODD as operating conditions under which a given driving automation system or feature thereof is specifically designed to function, including, but not limited to, environmental, geographical, time-of-day restrictions and the requisite presence or absence of certain traffic or roadway characteristics [6]. An example would be an autonomous commercial truck designed to operate in port, transferring goods in specified routes with a maximum speed of 30kmph during day time. It is up to the manufacturer of the autonomous vehicle to determine the vehicles ODD. This makes different original equipment manufacturers (OEMs) defining different ODDs for their vehicles. But in future, if we need a control tower to monitor AVs belonging to different manufactures, it will be required to develop a consensus among them in defining a common ODD baseline.

ODD elements

The ODD encompasses restriction on the road environment, the behaviour of the autonomous vehicle and on its state. Possible limitation on road environment are the type of the road (urban, rural, etc.), road surface friction, geographic area, etc. Examples of limitations on AV behaviour are the way of approaching intersection, in turning, speed limit, etc. Possible limitations on state of the AV are the maximum weight of goods to be loaded, tire inflation level, etc. All these restrictions constitute as ODD elements which need to be defined for the vehicle to operate autonomously. However, these elements can be limitless in number.

Czarnecki (2018a) defines a taxonomy of basic terms/elements used in the description of ODD for an ADS. Among other, the taxonomy defines operational world models which couples the ADS in its operating environment and terms for specifying driving scenarios and their attributes. His work as in Czarnecki (2018b) and Czarnecki (2018c) also covers operational world model ontology which is a conceptualization of all the ODD elements, its types, etc. for specifying operational world models for an ADS.

In this thesis, we use the ontology to get the list of possible ODD elements that the AV might come across during its operation. The elements are mainly classified into 5 root categories:

1. **Road structure:** This includes road types, capacity, surface types, road traffic control devices, pedestrian crossing facilities, cycling facilities, junctions, temporary road structures, etc.
2. **Road users:** This includes road user classification and road user behaviour.
3. **Animals:** This includes urban and wild animals categorized by their size.
4. **Other obstacles:** This includes remaining objects on the road way unclassified in other categories such as a lost cargo, objects placed by forces of nature, etc.
5. **Environmental conditions:** This includes atmospheric conditions, lighting conditions and weather related road surface conditions.

3
Research Method

This chapter discusses the applied methodology and methods in the study. The artifact is constructed using design science research (DSR) methodology in three iterations. The chapter begins with discussion on DSR with its main iterations and how they are executed in this thesis. It is followed with overview of different methods applied during the course of the study.

3.1 Design Science Research

Design Science Research is distinguished from routine design by the production of interesting (to a community) new and true knowledge (Hevner et al., 2010). It typically involves the creation of an artifact and/or design theory as a means to improve the current state of practice as well as existing research knowledge (Baskerville et al., 2018). It is a problem solving approach where knowledge is acquired during the process to generate an artifact as the end result. Given the control tower anticipation in Chapter 1, there exists apparent business value along with knowledge creation from the artifact. The study employs this methodology to create the artifact as summarized in Chapter 4, which is the list of use cases, its classification, list of challenges, requirements information model and the use case template.

The designing of artifact was a rigorous iterative process, swept across three iterations. This incremental approach provides continuous feedback which ensures that the artifact progresses accordingly. Each iteration encompasses different stages which vary across literature's (Dresch et al., 2015). We focus on 3 main stages: problem identification, solution design and evaluation. Figure 3.1 illustrates these stages with all the methods employed in individual stages. The iteration in which these methods are used is also highlighted within parenthesis.

The research questions are categorized into the stages of DSR and are answered subsequently by gaining knowledge as moving into further iterations. The artifact revolves around use cases and using this method, it will allow to better characterize the actual problem (RQ1), which is not as clearly visible from the beginning as the work in the domain is new. By brainstorming, literature review and discussions, potential mitigation strategies (RQ2) for the problems are found. Finally, the mitigation strategies are evaluated to measure the extent to which they solve the problems (RQ3).

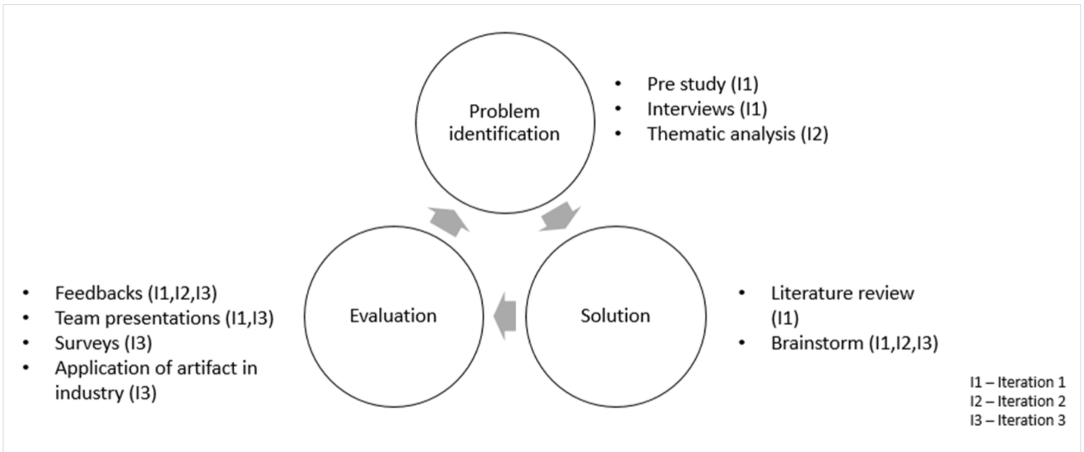

Figure 3.1: Stages of DSR along with the methods employed during 3 iterations

3.1.1 Problem Identification

The methodology begins with the problem identification, which involves investigation to find the relevant problem that should be solved. Notably, identifying a problem has particular relevance to DSR researchers, because they generally claim to solve real-world problems and, thus, need to be cautious to avoid type 3 errors; errors that occur when one uses the right research method to provide answers to the wrong question (Rai, 2017). To avoid this error, the first iteration should focus more on identifying the right problem rather than the subsequent iterations. For this purpose, a prestudy, interview, thematic analysis and review of feedback were the main source.

From the industrial proposal and subsequent prestudy, the problem was identified to be the need for important use cases of cloud based support for autonomous trucks and understanding of the relation of ODD elements and challenges in these prioritized list of use cases. However, during the interview with the cloud support team in the company and after thematically analysing the transcribed interview data, additional problems were identified. These problems amplified with the discovery of new problems during subsequent iterations and all the identified problems are structured in Figure 5.1.

Interview was the main method for problem identification, during the first iteration. Immediately after carrying out interview and transcribing them, the solution stage was entered from receiving insight into some of the problems. It was during the second iteration that a structured thematic analysis was carried out. The reflection from not strictly emphasizing on the problem space is, on carrying out a study in a company, the researcher will be self obliged to yield a minimum viable product instead of investing time into more detailed research of the problem space. This quick solution asserts the correct direction in which the study must progress but also lead to consequences in missing out to picture the entire problem space which affects time plan, additional effort spent in reordering the report, etc.

3.1.2 Solution Design

The solution design focuses on the core of the DSR: design and development (Hevner et al., 2010). The solution developed must meet the stakeholder needs and the identified problems justify the solution development in the form of artifact. Such artifacts are potentially constructs, models, methods, or instantiations (Hevner et al., 2010). The study incorporates list of use cases, its classification, list of challenges, information model and use case template as the artifact.

During the first iteration, using literature review, list of use cases and challenges were mined. An initial template was also developed. The list of use cases and challenges were on varying level of abstraction and were in large numbers (45 use cases, 37 challenges). Since it was evident that the use cases and the cloud applications can be limitless, context of the use case was identified. This allowed to condense most of the problems as elements of use case template. The introduction of context allowed the elicitation and discovery of new contextual requirements step by step by using partial knowledge about contextual requirements (Knauss and Damian, 2015).

In the second iteration, an initial classification of use cases was designed and the template was refined. The relation between ODD elements and control tower use cases was discovered during this stage. This discovery, made to reflect on including context as an element in the use case template. It was found that ODD elements affect the context of AV operation and a missing use case is probably due to not defining the context. Since there could be numerous context of application due to numerous ODDs, it was better thought to be left to the companies to decide on the context of operation by including the ODD elements in the template. During the final iteration, the use cases, challenges, template and classification were refined and the information model was introduced.

3.1.3 Evaluation

Only by evaluating the artifact's "utility, quality, and efficacy" can future work be coordinated accordingly (Hevner et al., 2010). The evaluation of the artifact in this study was a continuous process and it was not postponed to the very end of the iteration. Regular discussions with both the academic and industrial supervisors helped to validate the findings.

During the iterations, evaluation of the artifact was continuously performed from the point of it being effective both in the company and from research perspective. An observation from applying DSR in this study is, each of the three iteration are by itself constitute multiple small iterations, walking through the phases of problem, solution and evaluation. Designing part of the artifact is followed by feedback which facilitates understanding of the problem in the artifact itself and redesigning it to adjust to the main problem space with minor modifications.

At the end of the first iteration a presentation was held in the company with the discussion on the initial use case template that was developed. In the final iteration, much emphasis has been given to the evaluation of the artifact adhering to the guideline of DSR to rigorously demonstrate the usefulness of the artifact. For this purpose, apart from the feedback received from supervisors, other methods had been thought of. Initially, using the summarized user stories gathered from interviews were thought to be used as a source to evaluate the extent to which the study findings meets their needs. However, getting hold of all the interview participants was harder. Furthermore, the artifact solves problems beyond the ones emerged during the interview and spans to cover other problems as in the Ishikawa diagram (see Figure 5.1). So, for the final evaluation it was decided to carry out 2 surveys and presentations to the team in the company to gather feedback. One initial survey measuring how useful each of the use case and each of the challenge can be for the cloud support for autonomous trucks (i.e. the control tower) and second, a final survey to measure how useful the findings are in solving all the identified problems. After these 2 surveys, the applicability of the use case template in the industry was verified by documenting industrial control tower use cases in the designed template.

3.2 Interviews

By investigating on use cases, control tower and ODD, the ambition of conducting interviews in the company was for the better understanding of the cloud functionalities and the domain elements. As Kvale (1996) states, a qualitative research interview attempts to understand the world from the subjects' point of view. Interviews are important to access interviewee's interior experiences (Weiss, 1995). It served to uncover the problem space with respect to control tower development and the company's requirements to fulfill the support for continuous development.

The entire selection of interviewees followed purposeful sampling which was done by the identification of participants who were knowledgeable or experienced in the area and participants who were available and willing to participate (Palinkas et al., 2015). Each interview lasted for around 30 - 45min and the consent for data collection was received and information on preserving confidentiality was assured. The interview also involved video based elicitation (Zehe et al., 2016) during midway. This involved showing a video clip[1] of the similar control tower solution in the aviation domain, to simulate discussion. This video based elicitation was introduced after asking the participants on their view of how a control tower for AV should look like, in order to avoid any manipulation of participants opinions. After showing the video, question was asked on whether their expectation for control tower was the same as in aviation and how and what will be the expected differences in the functionality. This introduction of video was done for the better understanding of the concepts, to avoid any misunderstandings and to simulate thoughts and discussion from the participants.

Table 3.1 shows the list of interviewees according to the order they have been interviewed. Since this study has been carried out alone, the observer has been

[1] Saab Remote Tower (Accessed on 2020-05)

substituted with the interview being recorded. These recordings were played and transcribed immediately after the interview. This way of analysis not only provided a better understanding of the data but also facilitated in tuning the interview questions (see Table A.1) for upcoming participants to explore the unknown and effectively moderate further interviews. However, this interview question modification is done only in a smaller extent built on the knowledge gained from previous interviewee.

Table 3.1: Participants of the interview

Name	Main role	Additional role
Interviewee A	Systems engineer	Teleoperation unit responsible
Interviewee B	Systems engineer	Software architect for AV missions
Interviewee C	Software engineer	Algorithm development and optimization engineer
Interviewee D	Lead engineer	Requirements engineer
Interviewee E	Domain architect for AV	Technology advisor and research director
Interviewee F	Domain architect for cloud support	
Interviewee G	Consultant for control tower solution	
Interviewee H	Consultant for control tower solution	
Interviewee I	Project manager	

Sample size may be increased if ongoing data analysis leads the researcher to realise that there is an omission of an important group or type of person from the original sample universe, who should be added to the sample in order to enhance the validity or transfer-ability of the findings or theory (Silverman 2010). Initially, interviews were planned to be conducted with only the developers of the cloud support team. However, after 4 interviews, the problem was seen overlapping with the functional capability of the AV. It was necessary to understand the division of workload between the AV and the cloud support. Interviewee E was selected on this basis. Immediately after, Interviewee F of the cloud support was able to confirm the thoughts and enhance the validity.

In addition to selecting participants based on the provisional analysis of previous interviews, followed by the author's judgement as to what kind of person would help to cover the knowledge gap, the participants were also selected from the recommendation of previous interviewees who were knowledgeable from their years of working in the company and from the voluntarily willingness to share their knowledge. Interviewee G, H and I were selected on this basis. Interviewees G and H as consultants had previous experience working in remote air traffic controller. Interviewee G is into the development of teleoperation for the company which led to a demonstration by the interviewee of the prototype of the company's teleoperation control panel and AV simulator to the author after the interview. Interviewee I being the manager had an all round information of the control tower and had covered the business aspects of it. This lead to various insights on functionality of the cloud on increasing the operational capacity of the AV. Finally, the collected interview data was used to perform thematic analysis.

3.3 Thematic Analysis

Braun et al. (2006) and King (2004) argued that thematic analysis is a useful method for examining the perspectives of different research participants, highlighting similarities and differences, and generating unanticipated insights. Thematic analysis is also useful for summarizing key features of a large data set, as it forces the researcher to take a well-structured approach to handling data, helping to produce a clear and organized final report (King, 2004).

This method followed two cycles of coding, close to the coding practices as described by Saldaña (2015). Figure 3.2 shows the process of how the analysis has been carried out. Apart from the first and second cycle of coding, the process includes initial phase before the beginning of analysis and the conclusion phase afterwards.

During initial phase, the interview data was organized in Excel sheet, ordered by the questions. The recorded data was not transcribed as a whole that includes each and every article or literals. The end result from transcribing were cropped sentences without affecting the meaning conveyed by the interviewees. This step had been taken for the optimization of work, due to the following reasons: long interview with much data and study performed by sole researcher. After completing transcribing and organizing the data, it was re read to be completely familiar with. This also included few revisits to the audio data to clear arisen ambiguities.

The first cycle of coding initially involved attribute coding method. Here, basic information such as who the interviewee is, their role and also the date when the interview was held were noted, which would help in the later stages of the work on requiring contact with the interviewees for further information if on needed any. The cycle involved in-vivo method of coding to the fullest. In this method, literal terms used by the interviewees were taken as they are in order to grab the inherent meaning (Ex: third-eye, analyze ODD, etc.). Few of the codes were generated using descriptive method, which summarized each of the statement as a code. For example, as stated by Interviewee B: *"..the control tower should be able to monitor AV"*. This statement was condensed to yield Monitoring as a code. After the generation of initial set of codes, themes were created iteratively visiting the codes (Ex: Fleet

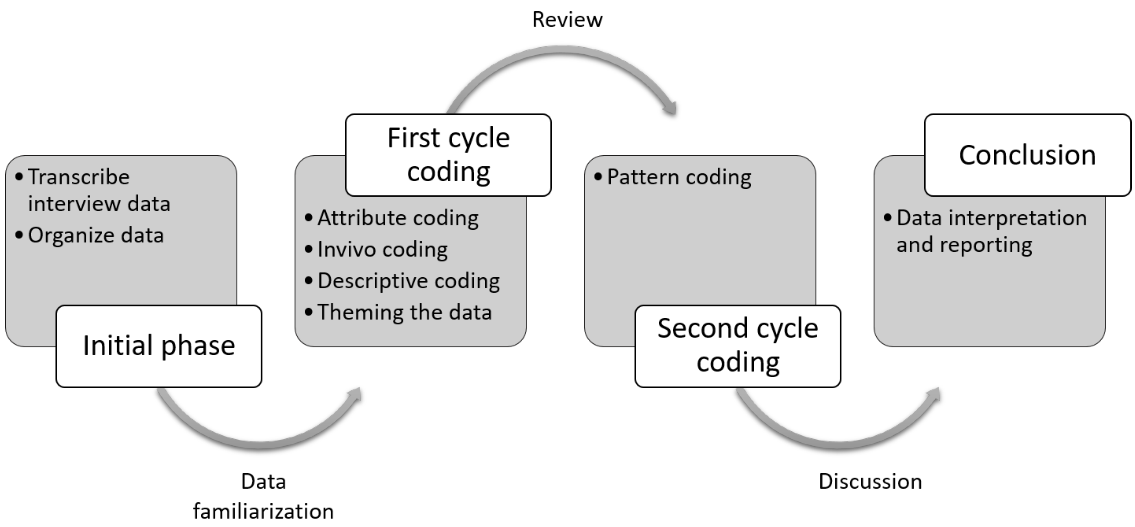

Figure 3.2: Method for conducting thematic analysis

management, ODD information management, etc.). To avoid bias in coding and to validate it, review was carried out with the help of the supervisor.

The second cycle of coding involved reorganizing and reforming the initial analysis result into more concrete and condensed form. This involved pattern coding where patterns in the data were seen to generate refined themes, whose connection to the main problem domain and to the requirements engineering was realized. For example, the code teleoperator appeared frequently in the data. Human factor is seen in connection to the code in requirements engineering, which lead to formulating the theme Human factor. The pattern coding was followed with discussions to finalize the themes. The final result was a set of 7 domain themes as seen in Table 5.1 and themes on functional requirement, quality requirement and information model.

As a final conclusion phase, since the thematic analysis was part of the problem identification, the analysed data has been interpreted and looked for main problems. The detailed result of this process is reported and discussed in Section 5.1.1.

3.4 Literature Review

Though not being a systematic literature review process, the process was conducted inspired by the guidelines provided by Kitchenham et al. (2007). The initial search in Google and Google scholar apart from providing the related articles also served as a base in finding necessary keywords and topics related to the search. The search strategy was designed and was applied on two of the main databases: Engineering village and Scopus. The search query for Engineering village was (“use case" or "services" or "challenges" or "problems" or "applications") and (vehicular cloud or control tower) and autonomous)". The search query in Scopus was TITLE-ABS("use case" AND "autonomous vehicle") AND ("vehicular cloud" OR "control*"). The end result was the list of use cases and challenges.

The immediate results from the search query (Phase 1) were filtered after looking for relevance in the title (Phase 2). This was further filtered based on the abstract and conclusion (Phase 3) with the final selection of articles after reading the content (Phase 4). The resulting process result is shown in Table 3.2.

Table 3.2: Literature search result

Source	Phase 1	Phase 2	Phase 3	Phase 4
General Search(Google and Google scholar)	-	-	-	11
Engineering village	77	26	13	4
Scopus	67	17	7	3

3.5 Surveys

The survey was created during the final iteration to answer RQ3 and validate the findings. As Pfleeger et al. (2001) describes, a survey is not just the instrument (the questionnaire or checklist) for gathering information. It is a comprehensive system for collecting information to describe, compare or explain knowledge, attitudes and behavior. This feature of survey makes it a good tool in our design science methodology to convey the opinions and knowledge based information in the evaluation section. During the evaluation phase of final iteration, 2 online surveys were created using SoSci survey[2] which included Likert-scale questions (Likert, 1932) and open-ended questions to be answered in a text. These surveys were held after conducting a pilot run.

The first online survey included questions to evaluate the usefulness of the collected final list of 25 use cases and 15 challenges. Appendix A.3 shows the questions asked in the survey. The respondents were the current team in the company who worked with the control tower solution. The respondents could type their roles and years of experience. Of the participants, 3 were software developers, 1 lead engineer, 1 system/software engineer and 1 manager. The experienced participant had about 20 years of experience working as a developer and 2 years specifically in autonomous vehicles. The manager had experience of 20 years and the remaining participants had experience ranging from 1 to 4 years in autonomous truck development. The other open-ended field was the comments box introduced in most of the pages. This

[2]SoSci Survey – the solution for professional online questionnaires

was to help the participants to give in their immediate feedback if they have any, which might otherwise increase chances of the immediate thought being lost after completing the survey.

During the course of this thesis, since the team was downsized to nearly more than half of the members existing before, the initial online survey was planned to be introduced along with a virtual presentation so that we can encourage the participants to complete survey to yield good set of data to draw conclusion. To get the complete response from all the team members, 3 presentations were held due to the overlapping availability of time for them. They were introduced to fill in the survey after mid half of the presentation and all the 6 people in the cloud support team including the manager have completed the survey resulting in a 100% response rate. This survey took on average 15min for the participants to complete.

The second online survey also lasted on average 15min and was to evaluate the extent to which the findings solve all the identified problems. Appendix A.4 shows the questions asked in the survey. It also included the findings of the study in the summarized form but are clipped to show only the questions for space reasons. Similar to survey 1, the respondents could type in their role, years of experience and feedback in the comment box. In addition to it, the respondents were asked to enter the company they work in as the survey was forwarded to developers/researchers outside the case company working with similar control tower solution. The survey had all the 6 team members in the company filling it after 2, short 20min presentation in the company. Among these respondents Interviewee C and D are involved. After this, the survey was emailed to other available participated interviewees. Of the 5 people for whom the survey was mailed, we had 2 respondents: Interviewee B and F, who both had above 20 years of experience.

The data gathered from both these surveys were downloaded, filtered and analysed using GNU R[3] and the results are presented in Section 5.3.

[3]The R Project for Statistical Computing

4

Artifact

The following chapter will present the artifact as a summary that has been designed from the study. The artifact presented here is a collective unit that includes the list of use cases, use case classification, list of challenges, information model and a template. These artifact units are preceded with a technological rule majorly through user stories summarized from the interview, in order to identify and communicate the knowledge part or the value creating aspects using DSR in this study. Aken (2001) describes technological rule as a chunk of general knowledge, linking an intervention or artifact with a desired outcome or performance in a certain field of application. The 'general' in this definition means that it is not a specific prescription for a specific situation, but a general prescription for a class of problems.

Section 4.1 shows the classification that emerged from the study which helps to organize relevant use cases in relevant categories. Section 4.2 shows the mined list of use cases and Section 4.3 with the list of challenges. The ID's used in listing these use cases and challenges are irrespective of any order in which the review has been performed. Section 4.4 shows the designed information model. Lastly, Section 4.5 presents the template followed with an example use case fitting the template.

4.1 Use Case Classification

The importance of the use case classification is summarized best in the following user story which was formulated from one of the feedback received during iterations:

US1: *"As a lead engineer, I want a standard classification that can be applied on a growing list of control tower use cases so that it would ease the task of maintenance of the use cases while leading the development of the solution"*

Along with helping maintenance, the classification is intended to resolve differences in the level of abstraction of the use cases and to easily prioritize a set of use cases. The use cases are classified into 6 segments as follows:

Improved traffic performance: It covers all the use cases that contributes to reduced traffic congestion, increase in traffic flow rate, minimum queue length, optimized traffic light, etc. that makes the drive operation of AV on road efficient.

Improved traffic safety: It includes set of use cases that improve the existing safety of the AV or introduce new safety functionalities through avoiding traffic accidents, prevention of collision with obstacle, etc.

Reduced local emission: Functionalities that contributes to ecodriving by reducing fuel consumption of the AV.

Increased individual needs: It focuses on all the use cases that proves beneficial to the society, local authorities, etc. It also includes use cases that provide benefits assuming a safety driver in the AV.

Enhanced data collection: Includes all the use cases that facilitate the cloud support by making the best use of the data collected from the AV and third party services like traffic authority, weather report station, etc. The segment considers use cases that function to deploy data as an asset for AV and societal benefits.

Other functionalities: It clusters use cases that enable regular operation of AV such as start up, shut down, maintenance check, health check, etc.

Table 4.1 shows this classification performed on the use cases which will be described in the upcoming section.

Table 4.1: Classification of collected use cases into segments

Segments	**Use cases**
Improved traffic performance	UC1, UC2, UC3, UC4, UC5
Improved traffic safety	UC7, UC12
Reduced local emissions	UC15, UC16, UC17, UC19
Increased individual needs	UC6, UC8, UC9, UC25
Enhanced data collection	UC10, UC11, UC13, UC14, UC20
Other functionalities	UC18, UC21, UC22, UC23, UC24

4.2 List of Use Cases

The importance of an initial set of use cases is summarized best in the following user story which was derived from interview data (see Section 5.1.1).

US2: *"As a lead engineer, I want to know use cases that are currently existing in the domain so that it would help to lead the development from the foundation overlaid by the use cases"*

To achieve this effect and to determine the wide range of cloud support functionalities the following list of 25 use cases are used:

UC1 Synchronizing traffic lights after clearing an accident (Stephan et al., 2011)
Description: The computational resources of all the vehicles will be used to recommend to a higher authority a way of rescheduling the traffic lights that will serve the purpose of decongesting the afflicted area as fast as possible.

UC2 Autonomous mitigation of recurring congestion (Stephan et al., 2011)
Description: The vehicles in the vicinity will be able to query the plan of each other at time of rerouting during congestion and estimate the impact on local roads to prevent vehicle flooding.

UC3 Efficient use of road space (Maheswaran et al., 2019)
Description: Implement allocation algorithms for efficiently dividing road space among competing vehicles. The lane configurations are customized to safely pack the increasingly heterogeneous traffic onto the road space.

UC4 Optimizing traffic signals (Whaiduzzaman et al., 2014)
Description: The vehicular cloud is sensing the segment traffic congestion by transferring the time, GPS coordinates and final destination to a navigation server through on-board vehicle navigators. The navigation server implemented in the cloud is in charge of computing the optimal routes by constructing traffic load map and the traffic pattern matrix and estimating road segment loads and delays.

UC5 Dynamic management of high occupancy vehicle (HOV) lanes (Stephan et al., 2011)
Description: Factoring data from sensors on board the individual vehicles, e.g. occupancy sensors, and local traffic intensity in order to optimally configure the HOV lanes.

UC6 Managing evacuation (Whaiduzzaman et al., 2014)
Description: The information collected through individual vehicles can be integrated together with information about traffic on a road available through inductive loops, cameras, roadside sensors and surveys to form a real time picture of the road situation.

UC7 Information sharing through an outside-the-car observer (Maheswaran et al., 2019)
Description: The observer can pre-process the data feeds obtained from the vehicles or other road objects before passing them along to other vehicles. For instance, pothole information or changing driveability conditions during wildfires, etc.

UC8 Detecting dead animals and other objects on the road (Aydin et al., 2018)
Description: Cleaning crew and police can get notified in real-time where exactly these objects lie. This can allow them to remove the objects very quickly from the road

UC9 Dynamic management of parking facilities (Stephan et al., 2011)
Description: By real-time pooling the information from the vehicles in the vicinity about the availability of parking at various locations inside the city, vehicles can be directed to the most promising location where parking is (still) available.

UC10 Weather report conditions and forecast (Lovas et al., 2019)(Marosi et al., 2018)
Description: Weather nowcasting shows very short-term weather predictions based on recent and localised measurements. Incorporating data, collected from a large set of vehicles allows these predictions to be more precise and detailed than traditional weather forecasting.

UC11 Traffic sign recognition data collection (Lovas et al., 2019)
Description: A statistical analysis on comparisons between existing traffic sign database and the one collected from vehicle can be used to identify new, changed, missing/stolen traffic signs and update the traffic sign database with the respective information. This database can provide useful information not only for the traveling users, but can also provide the road maintenance services with notifications where to check the road signs for dirt or if they were stolen.

UC12 Provide information about obstacles and blind spots on road (Kumar et al., 2012)
Description: A cloud-assisted system for autonomous driving can significantly improve the safety of autonomous driving, by providing vehicles greater access to critical information and blinspots.

UC13 Monitoring of accidents (Lovas et al., 2019)
Description: Vehicles with their array of sensors can perform as clients by collecting field data related to accidents. The spatial (map data) and non-spatial (sensor parameters) data is sent to a database center where it gets merged into one database. The main system analyses the incoming data to achieve various types of statistical reports.

UC14 Back office providing details of all the errors, warning, etc. (Marosi et al., 2018) (Lovas et al., 2019)
Description: A cloud-based framework that serves as one of the user interfaces for accessing the different functionalities of the platform, both for users and administrators.

UC15 Maintain a steady speed at low RPM (Lovas et al., 2019)
Description: Intensive braking and unnecessary speeding both wastes fuel, and increases safety risks. The analysis can help the vehicle with acceleration and speed graphs.

UC16 Shift-up early for information on improper shifting up tendencies of the vehicle (Lovas et al., 2019)
Description: With the help of the vehicles RPM and speed information improper shifting up tendencies and habits can be examined and corrected later on.

UC17 Automatic tire pressure check before drive, after and at high speeds (Lovas et al., 2019)
Description: Automatic checks can be performed frequently at least once a month and before driving, and at high speeds. Fuel saving can be estimated from this value.

UC18 Regular maintenance (Lovas et al., 2019)
Description: Collecting service information and recommending parts and service stations.

UC19 Cloud based algorithm that generates an optimal speed trajectory to reduce the fuel consumption (Ozatay et al., 2014) (Najada et al., 2016)
Description: The server generates a route to reach intended destination, collects the associated traffic and geographical information, and solves the optimization problem by a spatial domain dynamic programming (DP) algorithm that utilizes accurate vehicle and fuel consumption models to determine the optimal speed trajectory along the route.

UC20 Road condition monitoring (Lovas et al., 2019)
Description: The collected vertical acceleration data from the vehicle allows identifying road faults and for assessing the vibration load. These vibration values combined with collected trajectory and speed data can be used to profile routes according to road quality, goods vibration load, etc.

UC21 HD map generation/update (S. Liu et al., 2017)
Description: As the autonomous vehicles are moving and collecting new LiDAR scans, they compare in real time the new LiDAR scans against the grid map with initial position estimates provided by GPS and/or Inertial Measurement Unit (IMU), which then assists these vehicles in precisely self-localizing in real-time.

UC22 Running a complex battery lifetime model in the cloud (Adhikaree et al., 2017)
Description: Makes it possible to predict the state of health of high voltage batteries.

UC23 Plausibility checks (Milani et al., 2019)
Description: Manipulation in electronic control unit (ECU) software, sensor specifications or transmitted data can hamper the vehicle's normal operation. In order to verify the plausibility of sensors and ECU software, the models of vehicle software with series data and initial sensor specification are stored in the cloud and can be compared.

UC24 Remote engine start (Lovas et al., 2019)
Description: It allows to remotely start and run the vehicle for a definable period. This allows e.g., to control the interior temperature using the climate control system, and to pre-warm the engine.

UC25 Charging station managements (Herrnleben et al., 2019)
Description: Cloud connectivity provides fine-grained live data to optimize arrival and battery estimations and for using the vehicles as IoT sensors. Example of an optimization is the allocation of charging stations by booking an estimated time slot.

4.3 List of Challenges

The importance of the list of challenges have been summarized best as described in the following user story:

US3: *"As a systems engineer and teleoperation unit responsible, I want to know the potential challenges in the implementation of the control tower so that all technical issues can be noted early at the system level"*

Here are the list of challenges with a short one/two line description along with the reference to the literature from where it is extracted:

C1 Varying latency and delay time of cellular network (Milani et al., 2019) (Hobert et al., 2015)
Description: The transmission delay over the wireless link, delay induced by the security mechanisms (generation and verification of signature and certificate, respectively), etc.

C2 Data load control (Hobert et al., 2015) (Milani et al., 2019)
Description: Data load in the network gets amplified by high vehicle density corresponding to exchange in control message and data.

C3 Cellular network availability (Milani et al., 2019)
Description: It plays an important role in accessing the cloud based vehicle functions (CB-VF), e.g. in case of temporary lack of cellular coverage in a dead-spot zone, the on-board ECUs do not have access to the CB-VFs.

C4 Highly reliable packet delivery (Hobert et al., 2015)
Description: A lost or erroneous message might cause a malfunction of the vehicle control algorithms and create a safety risk.

C5 High message rate (Hobert et al., 2015)
Description: In the first generation of V2X communication systems (1G-V2X),

vehicles periodically broadcast safety messages with an interval between 100 ms and 1s. This will lead to high message rate as the monitored vehicles increase.

C6 Elastic mobile architecture (Stephan et al., 2011) (Whaiduzzaman et al., 2014)
Description: The architecture must be developed to accommodate changing application demands and resource availability on the move.

C7 Robust architecture (Whaiduzzaman et al., 2014) (Stephan et al., 2011)
Description: The architectural blocks must be designed to withstand structural stresses induced by the inherent instability in the operating environment.

C8 Shared responsibility (Maheswaran et al., 2019) (Milani et al., 2019)
Description: At times for the ultimate vehicle drive actions it is best to have inputs from both cloud and vehicle. The distributed characteristic of vehicle applications between cloud and on-board electronic control units require an appropriate partitioning.

C9 Managing highly dynamic cloud membership (Stephan et al., 2011)
Description: There is a critical need to efficiently manage mobility, resource heterogeneity (including sensing, computation and communication), trust and vehicle membership (change in interest, change in location, resource denial and/or failure).

C10 Privacy and security challenges (Whaiduzzaman et al., 2014)
Description: Includes data security, cloud access control, securing vehicular communication, securing vehicular communication, securing location information such as traffic status reports, collision, etc.

C11 Trust assurance (Stephan et al., 2011)
Description: Resolution of a problem such as on considering the workable schedule of the traffic lights produced by AV in a traffic jam, resides in some form of a trust relationship that needs to be forged between the municipal or country authority and the cloud.

C12 Effective operational policies (Stephan et al., 2011) (Whaiduzzaman et al., 2014)
Description: These are needed for seamless inter-operation, decision support, establishing accountability metrics, standardization, regulations, and even local and national policy making.

C13 AVC utility computing (Stephan et al., 2011)
Description: Need for economic models and metrics to determine reasonable pricing and billing for AVC services.

C14 Federation of different clouds (Whaiduzzaman et al., 2014)
Description: Interoperability of different types of clouds, connection, synchronization, and reliability and efficiency should be addressed.

C15 Sensing and aggregation data (Bitam et al., 2015)
Description: New research solutions are required to efficiently sense and aggregate various types of sensor data, including traffic data, vehicle's health information, information about the environment (disasters, fire, etc.), movements of vehicles and citizens on roads, etc.

4.4 Requirements Information Model

The information model is mainly designed to address the problem gathered from the interview as in the summarized user story below:

US4: *"As a systems engineer and software architect for vehicle missions, I want to know how the ODD elements can be effectively utilized so that it would aid in architectural decision contributing to optimized mission"*

Apart from this problem, thematic analysis of the interview data revealed themes pertaining to quality attributes. So, in order to achieve the above user story and to reveal the relation of the control tower use cases with quality attributes and other entities, the following information model is designed:

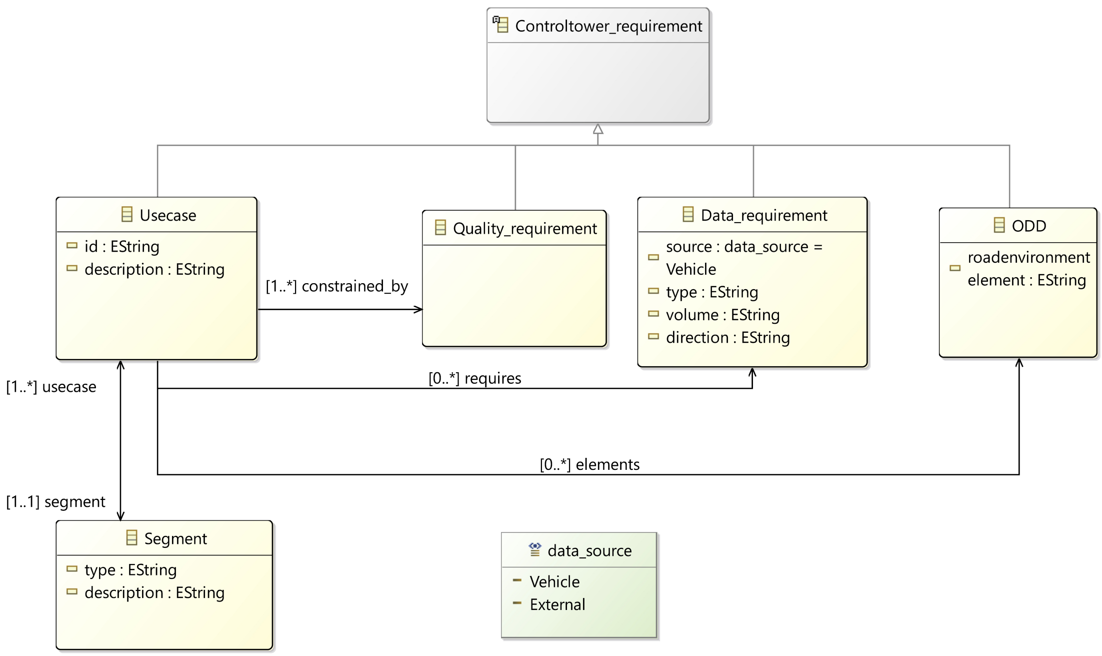

Figure 4.1: Information model showing the relationship of use case with ODD, quality attributes and other entities.

The key take away from this model is that the ODD that constrains AV, provides requirements for the control tower. Most of the use cases which are applicable during the operation of AV, includes the road environment elements. Identification of these elements and the impacting quality attributes can be effectively utilized in designing the various functional units of the control tower.

4.5 Use Case Template

The use case template through its elements is mainly designed to address the problems gathered from the interview as in the summarized user stories below:

US5: *"As a domain architect of autonomous truck, I want to know which of the use cases can be applicable to a fleet of vehicles so that flexible architectural decisions can be taken"*

US6: *"As a systems engineer and teleoperation unit responsible, I want to have information about the teleoperators so that an estimation can be obtained at the system level"*

US7: *"As a project manager, I want to receive information on possible customer value that can be imparted from the individual use cases so that effective project decisions can be made"*

US8: *"As a lead engineer, I want to know the data related information required to be collected from AV for each of the use cases so that it could aid in planning the software development and decisions across departments"*

In order to achieve the above user stories, to support emergence of new use cases by realizing the relation of the use cases with the ODD elements and to help to maintain use cases by including only the concrete details determined from the interview, the following template is applied:

Table 4.2: Use case template for control tower solution

Use case	<ID> <the name should be the goal as a short active verb phrase>
Segment	<name of the segment to which it belongs>
Description	<a one or two line description of the goal>
Fleet applicability	<Boolean>
Teleoperator requirement	<Boolean>

Expected benefit	<list of all the possible benefits and areas the use case can be helpful>
Quality attribute	<list of important quality attributes that apply to this use case >
Data requirement	<list of data required, data volume in bytes per hour, source and direction of data transfer>
Display principle	<the action that starts the use case>
Steps	Step Action 1 <steps of the scenario from trigger to goal delivery, and any cleanup after> 2 <...>
Existing solution	<list of software that already implement the use case>
ODD elements	<a hierarchy of all the ODD elements from the taxonomy that impacts the use case>

Table 4.3: Use case template example

Use case	UC8 Detecting dead animals and other objects on the road
Segment	Increased individual needs
Description	On detection of obstructing objects on road by the AV, the cloud is notified which can then take corresponding safety measures.
Fleet applicability	Yes
Teleoperator requirement	No
Expected benefit	Identifying the location of objects or dead animals so that police and clean-up crews are able to clean the road more quickly. Also contributes to road safety.
Quality attribute	For performance efficiency, the cloud and the AV must be able to reduce data latency.
Data requirement	location(longitude, latitude), dimension of the polygon(length, height, width), timestamp, where it is on the road (side, middle, right, left, etc.); 16TBphr; vehicle(source) to cloud (V2C).
Display principle	Trigger by the AV to update after analyzing change

Steps	Step	Action
	1	Cloud receives notification to update its HD map
	2	Updates HD map
	3	Notifies other AVs
	4	Notifies police or cleaning crew
Existing solution	Project Splatter	
ODD elements	Figure 4.2	

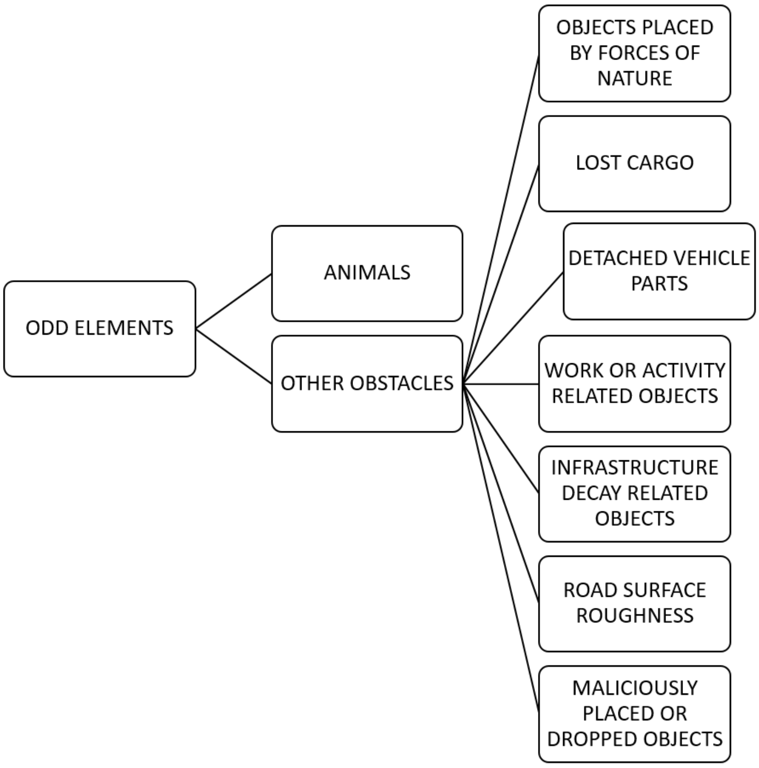

Figure 4.2: Hierarchy of ODD elements for UC8.

The listed ODD elements can give various insights and ideate creation of new use cases. In this example use case, the AV can identify what the animal is or the obstacle type. The hierarchy which has been developed from the ODD ontology (Czarnecki, 2018a) recognizes possible other elements under the root elements (here, animals and other obstacles). So, if the recognized other obstacle type object is lost cargo, then we get a new use case under improved traffic safety segment which functions to inform other AVs or other road users travelling along the same path. It can contribute to a new use case to improved traffic performance segment by helping to reroute on knowing the data and preventing congestion. Similarly, based on other ODD elements, new use cases can be ideated under different segments.

5
Findings

In this chapter, the three research questions stated in Chapter 1 are reported in detail of the findings. The chapter is structured to present results for each of the question separately. These findings are mapped to the three phases of the design science research: problem, solution and evaluation, with each of the phases reporting the collective results of the iterations carried out.

The chapter begins with the problems that have been identified, which is presented in Section 5.1. This problem exposure is followed by the solution candidates in Section 5.2, which describes the artifact in detail. Finally, the chapter ends with evaluation of the artifact in Section 5.3.

5.1 Problem (RQ1)

The section presents the problems identified mainly from the interview and from analysing the results after literature review. However, additional problems were discovered during the iterations which are also discussed in this section. Finally, all the identified problems are structured and visualized. The first research question is restated here to express the motive behind:

> RQ1: *Which problems are encountered when specifying use cases to support continuous development of off-board control system of autonomous trucks?*

5.1.1 Interview Analysis

The best way to understand which problems do the developers of the cloud support team face, was to ask the developers themselves. With this approach through interview, it was not just the technical requirements that would be elicited, but also subjective expectations in future when the autonomous truck will be fully functional on road. The transcribed data collected from the interview were analyzed using thematic analysis. Table 5.1 shows the final list of themes identified in the problem space and the listed codes under them. The table is followed by the explanation of each of the codes with example, closely following the practices by Saldaña (2015). The order of the description of the codes follows the order in which they are listed in the table.

Table 5.1: Domain themes and codes from the result of thematic analysis

Domain themes	**Codes**
Fleet management	Look after fleet Third-eye Online tasks Incident response Control unit Provide decision advice Long term planning Monitoring
Augment AV	Improve safety Identify risks Optimize AV Backup support Assist in rerouting Precheck before mission
ODD information management	Communication of information Analyze ODD ODD out of range
Human factor	Operator alertness and response time Knowledgeable teleoperator Number of operators Safety of personnel
Technical	Interoperation between control towers Connectivity between control tower and AV Connection to external services
Business	Providing customer value Transportation as a service Communication between different OEM's
Data modes	Data to be handled when AV is idle Data to be handled when AV is operational

Fleet management The control tower is focused not just on a single AV providing transportation as a service, but also with multiple AVs on road. Managing multiple commercial vehicles all over the road is a dynamic task and the vehicles being autonomous makes the process more dynamic and complex. Concern over fleet management as a problem has been reflected by several interviewees.

Look after fleet This code relates to observing the fleet and taking note of their operation. An example listed by one of the interviewee was Dynafleet application[1]. The cloud must support similar functionality to see where the vehicles are in real time, their charge state, etc.

Third-eye The cloud as a third eye can take into account, all the customer requests and the details happening in the road which the vehicles in the fleet would miss. For example, the traffic information on any congestion ahead on the road can be overseen by the cloud which can be used to suggest re-routing of the fleet to save time, etc.

Online tasks This indicates to shift the load from the vehicle to perform tasks online so that the computational capacity of them could be utilized effectively. Having a common off-board system for the fleet affects significantly the vehicles performance. An example could be that on detection of any accident on the road, the vehicle can notify cloud which would take the responsibility of sending information to the police instead of by AV.

Incident response Since the work is focused on trucks operating under level 4 autonomy, the vehicles are not capable to handle all the dynamic situations. In case of emergency, the AV just comes to a safe stop. For a single vehicle, having a temporary driver could solve this problem. However, this is not economically feasible when it comes to multiple vehicles. A cloud in this case, remotely can identify the vehicle state and can take suitable measures to handle incidents.

Control unit By sending missions to the vehicles, the cloud delegates responsibilities and sees to it that they perform the assigned tasks. By controlling the operation and delegating responsibilities to the autonomous vehicles, the cloud functions as a control unit.

Provide decision advice Cloud serves as an advisory unit to the vehicles. For example, when fleet becomes operational in a wider region, based on the obtained weather details, the cloud can provide advice to specific vehicles to reach the destination safely. However, it is to be noted that it is ultimately the decision of the vehicles to either accept or reject the advice based on their real time sensory based calculations.

Long term planning When there are any real time customer requests, it becomes challenging to decide which of the AV should me sent for mission. Various factors like fuel, maintenance, etc should be considered. There are also chances of selecting a vehicle which is already in operation to take the new mission. For the decision to be optimal, cloud should be capable to plan and predict for a long run.

[1]The Dynafleet app (Accessed on 2020-07)

Monitoring Example include monitoring multiple vehicles where one AV could be in the parking lot stationary, another AV in charging station, few in operation, etc.

Augment AV The theme covers functionalities of the cloud which try to enhance the existing vehicle capability by making it more safer on road and productive to the company. This is done by providing situational awareness if or otherwise the vehicle would fail to notice.

Improve safety For an AV, it would be hard to sense any object on the road in a turn or a cross over, if there are any obstacles in the direction of the turn. The cloud can then inform of any upcoming obstacle on the road and enhance vehicle safety.

Identify risks An example could be to design the functionality of the cloud to measure weather conditions so that, predictions could be made on upcoming extreme natural constraints and inform the AV.

Optimize AV Traffic coordination can be an important optimization problem. The cloud can act as an intelligent autonomous intersection management, by providing an optimal solution to avoid traffic congestion.

Backup support When there is a technology failure, the AV in level 4 autonomy comes to a safe stop. By providing remote backup computing resource in the form of cloud, that makes use of teleoperators, AVs can be assisted in operation.

Assist in rerouting An example of assistance in rerouting could be to use the information obtained by the cloud from traffic authorities to inform the AV of probable traffic jam on the route that it would otherwise choose to follow.

Precheck before mission It defines pre-checking of all the operation design domain elements on the route and confirming that they are within the limit of AV before sending it to a mission. This would make it productive as the chances of AV coming to a safe stop will be reduced.

ODD information management The operation design domain is a guiding factor for a vehicle to be autonomous. By sensing the elements falling in its domain on the route, AVs achieve self drive. ODD information management falls totally under the functionality of AV. However, it is possible for the off-board system to make use of the run time restriction of the vehicle to impact significantly on the vehicles.

Communication of information External ODD elements (weather, traffic situation, etc.) which might affect the service of the AV should be communicated by the cloud.

Analyze ODD The ODD capability on route is analyzed and filtered by the cloud. Decisions are then communicated to AVs.

ODD out of range The communication can also be 2 way, from the AV to the cloud. When the AV senses that there is an unknown obstacle on route not within its domain range, the cloud can be notified. It is then either an automatic data management or a manual action taken by the teleoperator.

Human factor Teleoperation will always be responsible for permitting the operation of the level 4 autonomous vehicles on road. Even though the control tower can be made autonomous for certain functionalities, teleoperator always comes in the loop. Factors like attention, human error, skilled performance, fatigue, stress, etc. are essentially critical from the viewpoint of providing an efficient transportation as a service Some of the factors reflected by interviewees are described below:

Operator alertness and response time This is mainly based on the cognitive functioning of the teleoperator. There might be times when the operator has no tasks to do which might lower the attention. The problem would then be on how to keep the operator alert and how quickly the incident should be handled by them.

Knowledgeable teleoperator In the hands of the teleoperator lies the safety of people on road. The teleoperator must be trained to handle various incidents, work in parallel and also be accurate with various controls in the dashboard.

Number of operators Though the company reduces its operational cost by eliminating the truck drivers, the net expenditure can remain constant with the number of teleoperators. It was seen from the interview as a problem on deciding the number of resource personnel required and how to make best use of the trained operators. With the growth of the number of control towers in future, there can also be research on whether a same teleoperator can be assigned work in another control tower simultaneously for optimal resource utilization.

Safety of personnel This is more related to the access to the control tower by authorized people. Safety in terms of surveillance is required when the teleoperator watches the autonomous vehicles.

Technical This theme addresses some of the technical challenges that the interviewees had mentioned. The main challenge was seen in network connectivity. Concerns were raised on seeing network issue as a major problem when the cloud solution scales in future.

Interoperation between control towers Successful operation of a single control tower might lead to development of many other control tower, which might also be situated in different regions. This arises need for coordination between control towers and a hierarchy of control towers with the parent tower to supervise child towers.

Connectivity between control tower and AV Mobility without connectivity can be hard to describe. The time-sensitive deliveries, safety related operation for which the teleoperation can be used, etc. relies heavily on connectivity.

Connection to external services Connectivity expands from the control tower and AV to control tower and external third-party services. For example, delay in receiving traffic information to the cloud could lead to possible chances of decrease in efficient use of road space by the truck. Delay in connection of the cloud to an emergency service like fire station could also lead to major consequences.

Business Many of the open questions about support for autonomous vehicles not just depend on the advancements in technical innovation, but also on the business models that emerge. From the interview with the manager, it was clear that the development of cloud support must also focus on how the solution will benefit the customer and how with that the company can attain profitability.

Providing customer value It is important to feed the AV with tasks that will create value for the customer. An example would be to modify the control tower to adapt to the customer needs specific to mines, if the autonomous trucks are operational in mining areas.

Transportation as a service Connected vehicles should simplify this service but handling of all the customer needs and ensuring the privacy of the customer data lies under the functionality of the control tower.

Communication between different OEMs The network of control towers might increase in future and it becomes necessary to formulate how different OEM's collaborate.

Data modes The concern over dealing with flood of data that might be generated by the AV was particularly highlighted by Interviewee D who was also simultaneously working on the data requirement. This theme highlights the capabilities of the control tower that are valid when it uses data and hence it is characterized as data mode.

Data to be handled when AV is idle Most of the time, AV can remain offline or not work(in safe stop). Chances are also that the vehicle might be in a parking slot. During this state, which of the data will need to be transferred to the cloud and how the cloud can best utilize it,etc. should be decided for the cloud.

Data to be handled when AV is operational Hundreds of sensors equipped in the AV can collect explosive data during operation, with the duration of operation being positively associated with the volume of data generated. Selected set of data can actually be utilized as an asset in improving the vehicle performance. For example, the road condition could be monitored by the data collected by the AV over a long period of time, which could be stored in cloud and not discarded.

In addition to the above domain themes, it was an interesting observation to discover 3 main themes specific to requirements engineering. The theme is defined and some of the codes under it are listed here:

Functional requirement As the name implies, functional requirement is a requirement that pertains to a functional concern (Glinz, 2007). The intended behaviour of the system that it must deliver is captured using the functional requirement.

The cloud support system was seen covering various remote based functionalities. With the thematic analysis the codes pertaining to functional requirements are fleet management, sending mission, assisting in traffic coordination, assisting in traffic queue, predicting traffic peak hours, rerouting based on extreme weather conditions, precheck of ODD before sending to mission, etc.

Quality requirement Quality requirement pertains to a quality concern other than the quality of meeting the functional requirements (Glinz, 2007). The codes that highlighted this theme are safety of teleoperators, increase of AV productivity, increase of vehicle safety, data bandwidth problem, data security, gdpr compliance of the collected data, etc. The increase of AV productivity relates to the throughput attribute of performance requirement. It is evident that there might exist some form of relation between these attributes and control tower use cases. The information related to it requires to be clearly established.

Requirement information model During thematic analysis, codes such as unsure how ODD affects and pre-estimation of ODD was seen as a commonality between the interviewees. The provided answers on the effect of ODD were also limited to only the traffic and weather condition details. Existing literature's lacked information on the functionalities in cloud that consider the run time elements of the AV. These factors contribute need towards information model.

From the analysis of interview data various scope for research were seen. However, the focus of the study will be restricted in providing effective use cases for the continuous development of a single control tower rather than focusing on multiple, hierarchy of such. The scope of this study will not expand in researching on practices for collaboration between OEMs and business models. From the result of the thematic analysis and the restricted scope, the summarized problems that must be addressed are listed below along with the themes that helped to discover the problem space.

The cloud support use cases of autonomous trucks to be effective must:

- cover various functionalities (theme: Fleet management, Augment AV, Functional requirement)
- check for fleet applicability (theme: Fleet management)

- provide information model of associated ODD (theme: ODD information management, Requirements information model)
- include information regarding teleoperators (theme: Human factor)
- cover technical challenges in the implementation (theme: Technical)
- specify customer value/expected benefit (theme: Business)
- provide information on data requirements (theme: Data modes)
- cover important quality attributes (theme: Quality requirement)

5.1.2 Literature Review

As an initial step to look into the problem, use cases on the existing work were mined from the literature (see Section 4.2). This list of use cases were initially defined as a solution to provide use cases for the company. But from undertaking this step, it was not just a solution to cover various control tower functionalities, but also gave rise to new problems which any developer might face if opting for continuous software development. The result, apart from providing solution, was an addition of two other problems to the existing problem space: difficulty in resolving the level of abstraction in the use cases and the emergence of new use cases.

A main observation from the collected list of use cases was the difference in the level of their abstraction. Since different authors had specified these use cases in their defined context, the collected list can inherit the varying abstraction. But this is also a problem in continuous development. It is not necessary that the same requirement engineer who had specified use cases initially might specify during rest of the iterations. It is also to be noted that deciding on the level of abstraction can be difficult and depend very much on the number of requirements and their complexity (Berander et al., 2005).

This difference of abstraction besides from causing difficulty in use case clustering also caused ambiguity during the development of use case template (see Section 4.5). At one point during iteration, contextual requirements were studied and was found applicable to one of the use case. A clear description can help to understand the use cases but when they outgrow in number, we need a proactive solution that clearly refines the use cases.

The other problem is seen as a need for support to help the developers in the ideation process to elicit use cases. This can be helpful especially during continuous development to release new features. It was seen from the literature's that since the control tower scope is not yet clearly defined and a large number of ODD elements exists there is room for explosive number of unknown functionalities. This in long run might lead to difficulty in maintaining them. So we need a solution must take into account to not only support the specification of new use cases but also help in maintaining them.

5.1.3 Additional Problems Identified

During the course of iterations using DSR methodology, the problem space will be revisited after evaluation from the previous cycle. The additional problems described here are the ones that were identified during iterations through feedback received from the academic and industrial supervisors.

One of the problem identified was the need for the prioritization of use cases. The notion of releasing progressive versions and updates on products, as well as the rising demand on developers to build systems that go to market much faster has led to the need to prioritise requirements at the earliest stage. Having a prioritized list could solve the problem as it will support the development of requirements in continuous order. To solve prioritization, the use case classification (see Section 4.1) was proposed. However, during this process it was found hard to determine the high level segments that remain stable as the use cases out grow in number, so that they can be maintainable.

During the first cycle when the initial solution was designed, strong connection of the use case as a contextual requirement was observed. With the help of defining a context, all the solution candidates were able to be captured and structured down. Though during later stages it was hard to relate to contextual requirements, it was found interesting to consider this as a problem and see how the use cases are connected to context.

The designing of artifact can also be challenging. Artifacts that contain more detailed information are also the ones on which people rely and are maintained. In our case, the problem lied in how to get the use case template (see Section 4.5) to be sufficiently detailed so that the use cases can be maintained.

The identified problems from the interview analysis, literature review and additional problems discovered during iterations are summarized in Figure 5.1. The ones from interview are suffixed with '*' sign and after the literature review with '+'. The non suffixed sub problems are from the additional identified problems subsection. The problem is structured using the template presented by Ishikawa (1990), commonly called fishbone diagram. In the figure, the identified problems are represented as sub causes to effect the root problem, which is the text box on the right side of the vertical dashed arrow. The 4 text boxes on the left side of the dashed arrow, group the identified problems and are represented as the main causes for the root problem.

Some of the problems shown in the Ishikawa diagram are more specific to the domain (no information on the applicability to fleet available, information on teleoperators required is unknown, etc.) and remaining are more general (difficult to determine wide range of cloud support functionalities, challenging to design the artifact to be adequate, etc.), which are found applicable in other areas as well. Since the problems that are generic are mentioned by the interviewees in this study context, they are taken into account for searching solutions. In Figure 5.1, problems that are specific to the domain are distinguished with a caret character than the rest

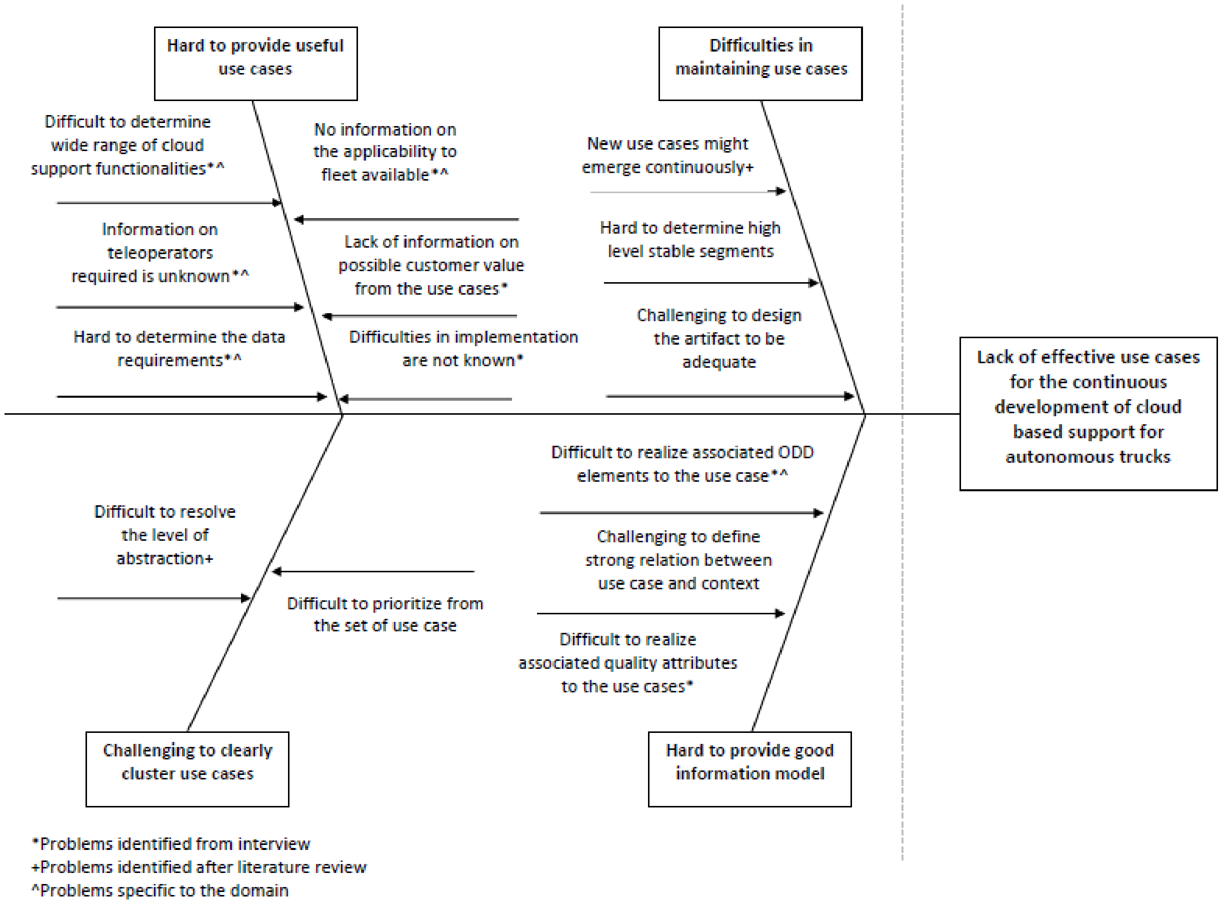

Figure 5.1: Ishikawa diagram representation of the entire problem space

of the problems. The selection of specific problems has been made considering the interview data, strengthened from reasoning out the discussions carried out with the company representatives during iterations.

5.2 Solution (RQ2)

The section presents the artifact summarized in Chapter 4 in detail, reviving the motive for each of it (list of use cases, use case classification, list of challenges, requirements information model and use case template), their key characteristics and stating why they are important in terms of the problems that they solve. User stories, already presented for each of the solution are elaborated to express what led to the intention to formulate the specific solution. The second research question is restated here to express the motive behind:

RQ2: *Which solutions can mitigate the problems identified in RQ1?*

5.2.1 List of Use Cases

Interviewee D had found it extremely important to know the existing/related initial set of use cases. It is so that, the team can use it to set foundation for understanding the possible functionalities to be included in the cloud support, how it could be developed, etc. US1 summarizes the interviewee's motive:

US2: *"As a lead engineer, I want to know use cases that are currently existing in the domain so that it would help to lead the development from the foundation overlaid by the use cases."*

In addition to it, as an initial step before development, it is important to investigate the existing solutions and the vehicle capabilities. With different OEMs involved in releasing autonomous trucks on road and equally or more advanced research done on the cloud solutions, it was found valuable to search on related work.

During the initial search 37 use cases were collected. However, during later iteration, they were described concisely after revisiting the referenced papers again. Attempt was made to crosscheck while writing the description to see if the least abstract use case was included in the corresponding higher abstraction level use case. If so, the least abstract use case was removed. The end result after filtering and from feedback was 25 use cases as listed in Section 4.2.

Key characteristics

The provided list contains high level use cases. The high level use case is simply a summary description of the task, written as unstructured text a paragraph or two in length. It does not include detailed structured description. Instead, each of the use cases are uniquely identified followed with a short description of its intended functionality. Each of these also include the reference to the literature from where they were extracted.

Importance

Use cases are generally analysed and formulated along with the design team. It requires several meetings and is developed in phases. Each use case development requires in itself to pass over multiple steps in terms of identifying the actors, selecting an actor, identifying the basic course, alternate course, etc. The provided list of high-level use cases, would partially lessen this overload and makes it possible to determine the existing cloud support functionalities and understand the system scope.

5.2.2 Use Case Classification

Clustering of related use cases would improve the development planning, as by achieving the right level of use case granularity which eases communication between stakeholders and developers. Having a large number of use cases causes difficulty to read and maintain them. It was one of the feedback received from Interviewee D during a meet after conducting interview, that a structured organization of the use cases will simplify maintenance and not cause problems during development. US1 summarizes the motive below:

US1: *"As a lead engineer, I want a standard classification that can be applied on a growing list of control tower use cases so that it would ease the task of maintenance of the use cases while leading the development of the solution"*

When large number of use cases accumulate, they should be prioritized to be implemented in successive builds. This task could be tedious, given the volume of use cases. Jacobson (1993) points out that the specification of 5-20 use cases should take 3-6 months. Specification of the collected list of use cases and then prioritizing them is quite challenging given the time constraint. If the prioritization had to be carried for the use cases, then the solution would be to either ask the customer opinion or go by the engineer's previous experience (Moisiadis, 2000). An approach would be to perform classification so that we can just prioritize the high level requirements, and then can let the subordinate requirements inherit the priorities (Wiegers, 1999).

During the literature review, no significant clear classification was observed other than having random application based classification such as traffic management, parking management, etc (Whaiduzzaman et al., 2014). The review process also revealed classification of cloud based vehicle functions according to the application model, as functions running only on cloud, elastic, duplicate on both cloud and the vehicle, etc (Milani et al., 2019), but not the classification of use cases themselves. In this study, an initial classification with the collected use cases was made by grouping based on data (AV, cloud and in both) and another classification based on use cases in safety, convenience, ecodriving, etc. But none of these attempts seemed effective and it raised concern whether the classification segments can be stable enough when many use cases accumulate.

However, in the currently running innovation project[2], the benefits from having a control tower have been classified into 12 segments. This is the result from a workshop, were 73 ideas were brainstormed. 5 of the segments were related to traffic (Improved Traffic performance, Improved Traffic safety, Enhanced traffic information, Enhanced traffic optimization and New traffic planning prerequisites), 2 segments on productivity(Improved mobility services and Energy efficiency), 1 segment on data (Enhanced data collections), 1 on environmental benefit (Reduced local emissions), 1 on customer need (Increased individual needs) and 2 on accessibility (Prioritized accessibility and Increased accessibility). There is no information provided on the generated 73 ideas and there is lack of description on the meaning conveyed by the segments. Taking this classification as an inspiration, the collected list of use cases have been classified into 6 segments as seen in Section 4.1.

[2] AD Aware Traffic Control (Accessed on 2020-07)

The 6 segments have been defined in this study and the collected list of use cases are classified into these defined segments. The usage of segments from the project is influenced from the belief that they can be stable in nature as it is the result from carrying out the workshop with notable participants. The selection of the 5 segments out of the 12 has been influenced by two reasons: The understanding of the segments associated with available description and secondly, the list of use cases being able to cluster under the selected segments. Since the collected list of use cases were from range of papers, we can also say that, the remaining 7 segments mentioned in the innovation project could just be redundant. This is also supported by the fact that, the remaining segments had fewer count of ideas reported under them.

Key characteristics

The proposed use case classification as a solution follows the terminology used by Cockburn (2001). Here, he mentions classification of goals into summary goal (cloud and kite), user goal (sea) and sub function (fish and clam). The sub function are too detailed (adding a new field using dynamic HTML, update screen, set destination, etc.) and are not part of this study. Figure 5.2 shows the level of classification.

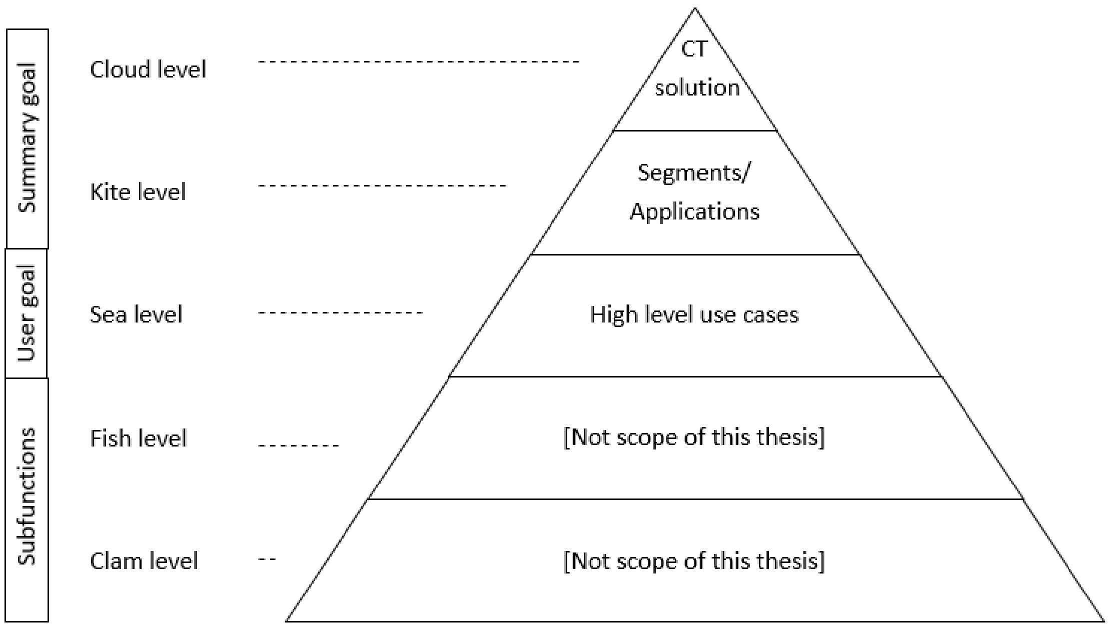

Figure 5.2: Classification according to goal level hierarchy

The user goal level includes all the main functionalities that must be achieved. Anything below user goal, i.e, sub function level includes all the system design specifications in detail. The provided high level use cases have been represented at the sea level. This is because detailing these high level use cases in the proposed template provides sufficient detail but not in depth information through complete task description, variants, etc.

The summary level shows the whole process in the context of the life-cycle. Recognizing summary use cases can be a valuable aid while determining the high-level requirements, but will not provide the functional requirements. In our case, it includes the 6 segments at the kite level which will help to navigate through the different applications that can be provided with the cloud based support. The ultimate cloud level represents the final control tower (CT) solution, that implements the underlying applications.

Importance

This classification helps to resolve abstraction level among the use cases. It makes it easier to prioritize use cases by prioritizing among small number of segments and letting the use cases underneath to inherit the priority. It was believed to be hard to determine high level stable segments for a classification. But since the segments are based on look through of the literature's and selecting from the literature where the result has been based from carrying out workshop with participants having similar knowledge, selecting the high level segments with clear definition and verification by actually clustering the collected use cases, makes the classification reliable and practically applicable.

5.2.3 List of Challenges

Reliability of the network and latency can cause teleoperation to fail at times. However effective the developed solution is, it will fail if there spikes network related challenges. This issue was mainly addressed by interviewee A who had witnessed this problem during initial prototype demonstrations and is stated in user story below:

US3: *"As a systems engineer and teleoperation unit responsible, I want to know the potential challenges in the implementation of the control tower so that all technical issues can be noted early at the system level"*

The implementation challenge remains unknown until the solution is completely developed. The code related issues faced by the developers can be resolved during the development. However, when the solution is integrated on a continuous basis, situations might arise that point the challenges emerged, to the decisions taken by the system architects at the pre-phase of the development. So it will be helpful to know potential challenges during the requirement stage itself. Section 4.3 shows the list of challenges mined from the existing literature on the related work.

Key characteristics

The collection presents challenges pertaining to architectural, functional, organizational, fleet operation, network based, security and privacy related issues. Similar to the list of use cases, the collection includes the reference to the literature, from where it was extracted. Challenges addressed by different authors that are similar in description are clustered together into a single challenge.

Importance

Providing this list, solves the problem of the challenges in implementation being unknown. Although the main challenge raised was on network connectivity, the solution addresses all the potential challenges associated with the implementation of cloud based support for autonomous vehicles.

5.2.4 Requirements Information Model

During interview, the highlighted ODD elements were weather and traffic situations. But the need for understanding the relation of ODD elements and control tower was seen crucial. The following user story summarizes the motive of one of the interviewee:

US4: *"As a systems engineer and software architect for vehicle missions, I want to know how the ODD elements can be effectively utilized so that it would aid in architectural decision contributing to optimized mission"*

Czarnecki (2018a) has provided terminology of all the ODD elements critical for the vehicle. A search on the existing literature's failed to provide information on clear relation between the ODD elements and the cloud support functionalities. As Wohlrab et al. (2020) mentions, requirements information models (RIMs) are artifacts that describe (1) entity types of information and concepts related to requirements engineering, (2) their relationships, and (3) constraints to create requirements-related knowledge. To provide the missing information, RIM was seen as a solution.

The model as seen in Figure 4.1 has been designed inspired from RIM as in Wohlrab et al. (2020) and the contextual requirement model from Knauss, Damian, et al. (2016).

Key characteristics

Here, we describe in detail the entities of the model and their relationships. *ControltowerRequirement* is the main entity. *Usecase*, *QualityRequirement*, *DataRequirement* and *OperationalDesignDomainRequirement* are specialized entity types of *ControltowerRequirement.* The entity *Usecase* can be constrained by *QualityRequirement* and the relationship of mainly the *Usecase* with other entities is highlighted in the model.

The entity *Usecase* represents a high level use case with a unique primary attribute ID and a textual description. The use cases are constrained by the quality attributes.

An example would be, the action to be taken by the cloud support on detecting obstacles on road. This must be achieved say within 3s on receiving data from the vehicle. Here, the performance attribute imposes limits on the use case implementation. This example also connects use cases to the entity *DataRequirement*.

The entity *DataRequirement* represents requirements that the cloud support must satisfy regarding data which the cloud support either receives or transmits. The source, direction of data flow, volume of data and the different types of data form the attributes of this entity. External services like traffic authority and weather stations and the vehicle are the two sources where the data is either received or transmitted by the cloud support. The entity *Usecase* requires *DataRequirement* as the cloud support is located remotely and is functionally supported by real time data for most of the vehicle functionalities.

The entity *OperationalDesignDomainRequirement* provides requirements for control tower such as the control tower must be alerted when the weather turns worse, when the ODD boundary for vehicles is no longer valid (vehicle in safe stop), etc. Since the term ODD also defines the state and the behaviour of the vehicle such as having no trailer attached, etc., the attribute elements are explicitly stated to indicate road and environment elements in the operation domain of the vehicle. The use cases in turn formulated on the foundation of none or many elements encountered in the operation domain of the vehicle. For example, in the use case regular maintenance (UC18), the data collected from the AV in the cloud data storage class are about the health status of the components of AV and not of the environment, accounting for no road elements.

DataRequirement is also a part of the *OperationalDesignDomainRequirement*. Considering the same example of detecting obstacles as in above, an obstacle is an ODD element and details of this element such as the object size, image pixels, etc. constitute the data requirement of the control tower. However, for not to complicate the model and nor the scope of this work, only the main relation between use case and rest of the entities are shown in the model.

Importance

To understand the ODD and quality attribute relation with the use case, the information model is helpful. The model sheds light on the information that the ODD that constrains AV, also provides requirements for the control tower. Identification of ODD road environment elements and the impacting quality attributes in a use case can be effectively utilized in designing various functional units of the control tower.

5.2.5 Use Case Template

A use case may be visualized as a use case diagram or/and in structured textual specification format. When the use cases get documented in natural language and scale in number, there arises incompleteness, insufficiency, cross-cutting, redundant and ambiguous use cases in the specification. As a result, a template serves the need by providing a standard representation. It is to be noted that the quality of the documented use cases is extremely important for the quality of the resulting software product (Anda et al., 2009). We achieve this by including only the template elements that can be useful, which is determined from the interview.

Following are some of the user stories that motivated the selection of template elements. The user stories are followed with a short description.

US5: *"As a domain architect of autonomous truck, I want to know which of the use cases can be applicable to a fleet of vehicles so that flexible architectural decisions can be taken"*

Each of the use case should be studied if they are related to fleet control unit. Knowing before hand that a use case can be applicable for fleet of vehicles or not can help to design and allocate fleet control software unit.

US6: *"As a systems engineer and teleoperation unit responsible, I want to have information about the teleoperators so that an estimation can be obtained at the system level"*

Teleoperators provide remote assistance in real time for autonomous vehicles. They replace the human drivers from behind the wheel to a centralized location/control tower, so that one person has the potential to assist many different vehicles across the course of the mission or during a day. In future, this remote service system asks a great deal of its remote operators and it becomes critical to estimate the number of operators needed with efficient management of their work hours in relation to the cognitive load. Daw et al. (2019) have developed a numerical method to compute the exact staffing level needed to achieve various performance measures. For this work, the focus will be on realizing if any of the cloud functionality requires a teleoperator or not.

US7: *"As a project manager, I want to receive information on possible customer value that can be imparted from the individual use cases so that effective project decisions can be made"*

Interviewee I was keen on understanding the value or the benefit the customer could get from the cloud support functionalities. This would add to business value and help to make decisions in the management level. Along with this, continuous development comes with delivering customer value at high speed. By having foresight on what the customers need and what value/impact a feature for the cloud support development could lead to better risk management of the requirements. Since the main terms identified during interview were on increasing the safety and the productivity of the vehicle, along with this, the social aspect on how the other people can be benefited, will be important when describing the use case.

US8: *"As a lead engineer, I want to know the data related information required to be collected from AV for each of the use cases so that it could aid in planning the software development and decisions across departments"*

Huge terabytes of data get generated from the autonomous vehicles as they operate in real time. For the cloud to guide the vehicles, it will be important to have real time transmission of vehicle data to the cloud. For it to function effectively, it will be helpful to know which data will be required from the vehicle for each of the functionality.

By the end of the interviews, the problem, divided into sub problems were broad and scattered. In order to structure it, the existing templates were not found applicable for the high-level use cases specific for the current problem. This motivated for designing a template following a formal description inspired from the one column table style of Cockburn, 2001. Considering the investigated problems, the proposed template has been formulated to ease the developers in terms of challenges, data requirements, etc. During iterations, modification to the template have been made. For generalization, the template has been applied to three other use cases too. The draft template along with example can be found in Appendix A.2. The final template after completion of iterations and an example of applying it to one of the collected use case is shown in Section 4.5.

Key characteristics

The final template consists of 12 elements. *Use case* along with an unique identifier, *segment* name under which the use case can be classified, *description* that describes the use case in a line or two, *fleet applicability* that checks whether the use case can be applicable in a fleet or not, *teleoperator requirement* which is again a binary value to check if the use case at a point might require teleoperators or not, *expected benefit* which describes the customer value that could be provided, *data mode* which indicates the type of data, the direction of data transfer and the data volume and *quality attribute* which lists only the important quality attributes applicable to the use case. According to ISO/IEC 25010 (2011) standard there are 8 quality attributes with 31 sub attributes. Some of the quality attributes are standard to the system and there is nothing special in them. So only the important quality attributes are listed here.

Apart from these 8 elements, the template includes two of the elements that are generally found in other existing templates: *Display principle* and *Steps*. The *Display principle* indicates the trigger for the start of the use case and the *Steps* is a step wise list of all the actions that take place in the use case from the trigger till the end of it. It will not describe the steps in detail but only a black box description of the intended system flow.

As an addition to the template after receiving feedback from the first cycle, the unique element *existing solution* is added. It mentions existing software or reference to resources that are similar to the use case. *Challenges* were initially added as an element to the template to include the ones from the collected list of challenges. However, the challenges seemed more applicable to control tower as a whole rather than an individual use case basis. So it was decided to be removed.

Finally, the template includes *ODD elements* that displays the hierarchy of associated odd elements. Identification of ODD elements helps to encourage the developers into thinking about all the possible ways the functionality could be provided which might otherwise trip off. From the knowledge of associated ODD elements in an existing use case, new use cases may be generated. A use case consists of 0 or more ODD elements (see Figure 4.1). By identifying all the elements listed in the ontology, starting from the root ODD category, we can create new use cases. Figure 5.3 shows this dynamic process.

Importance

The provided template is unique and to the investigated problem. Without this template, it would had been harder to resolve multiple causes and sub causes that make the problem. The template solves the check for fleet applicability, teleoperators, customer value, important quality attributes and data requirements. Emergence of new use cases is supported through ODD hierarchy which is of critical value to have new incoming use cases for the continuous development. The template is detailed yet concise enough which makes it easily maintainable.

5.3 Evaluation (RQ3)

The artifact was evaluated with the academic supervisor, industrial supervisor and with the (previous-)developers within the cloud team in the company. This section presents findings of the two evaluation surveys that have been carried out. To review the motive behind, the third research question is restated here:

> RQ3: *To what extent can the problems (RQ1) be solved by potential solutions (RQ2)?*

5.3.1 Initial Survey

The intention behind this survey (see Appendix A.3) was to evaluate the usefulness of individual mined use cases (see Section 4.2) and challenges (see Section 4.3). The results of it are presented in Figure 5.4 and Figure 5.5 respectively. Use cases are represented along with their segment ID and challenges with their ID's in the vertical axes of the figures. From the basis that the entire team working on control tower has evaluated the findings, we can argue on the strength of the results of the survey. However the results are specific from the evaluation in a single company.

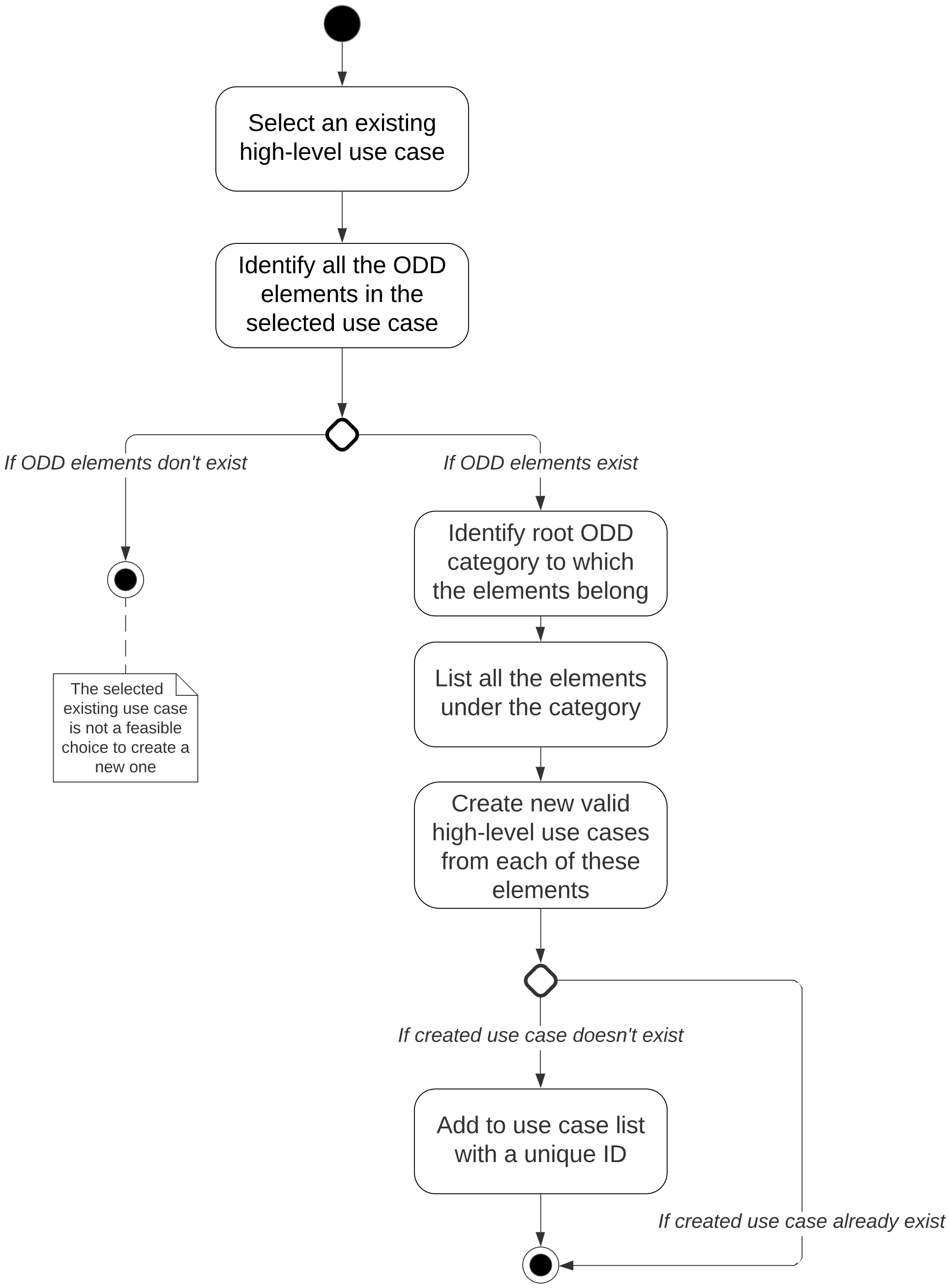

Figure 5.3: Activity diagram showing the process of creation of new use cases from the existing using ODD elements

As seen from Figure 5.4 more than half of the use cases seem to be helpful for the team. Use cases that received responses to be only "agree" and "strongly agree" were found to be the ones described technically and focused towards optimization algorithms, estimation, etc. The use cases that received responses in lower scale were found general and focused on assisting in the maintenance and regular operation of the vehicle.

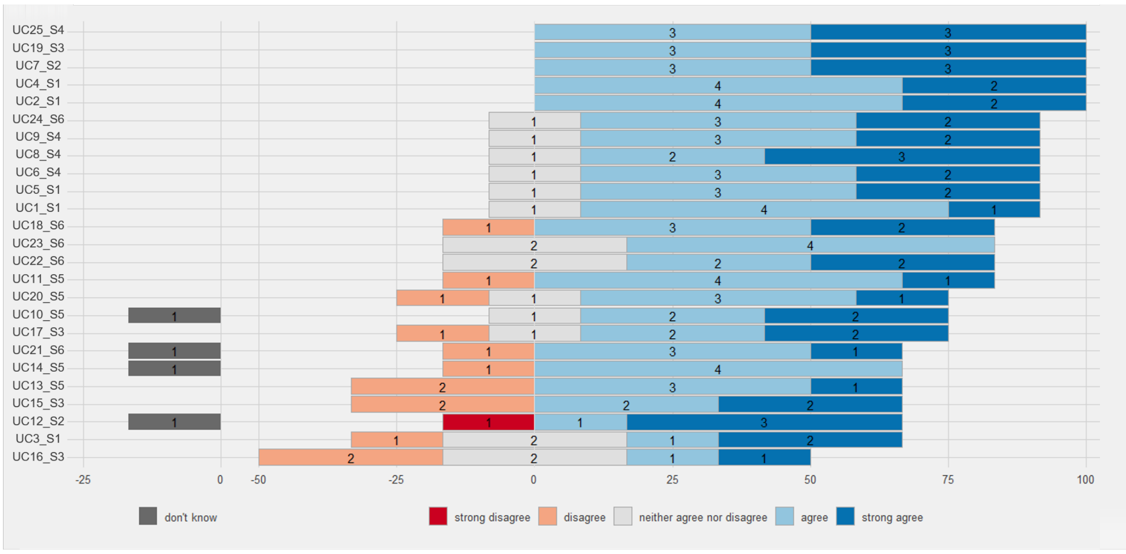

Figure 5.4: Results on usefulness of use cases for control tower development

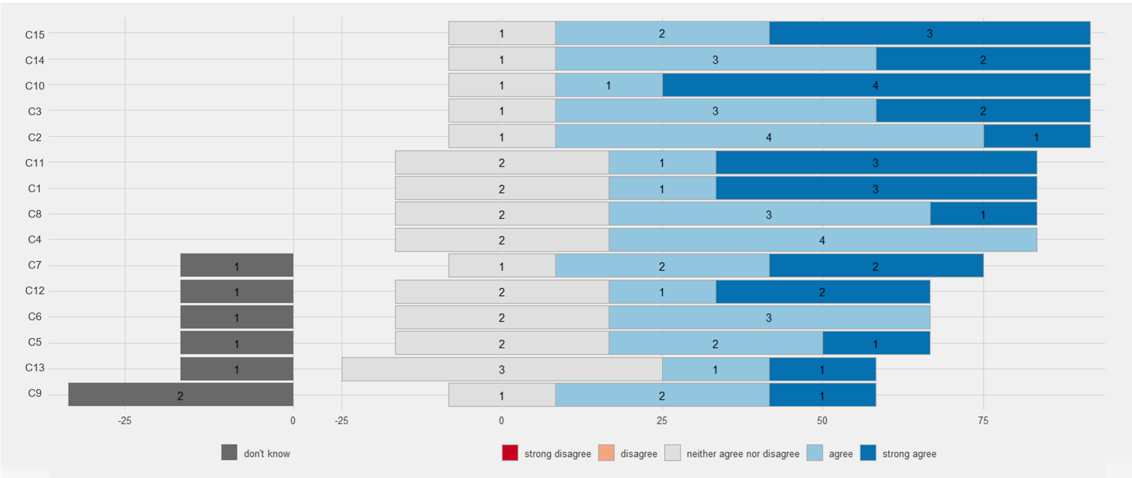

Figure 5.5: Results on usefulness of challenges for control tower development

One of the comment received was that few of the listed use cases can be applicable even for manual trucks. An example provided was regular maintenance (UC18) and plausibilty checks (UC23). A suggestion provided later during discussion was to further narrow the use cases, not including the ones that can be applicable to manual trucks. The reasoning behind this is, the work-load of the control tower should be minimum as the autonomous capability of the truck should make it fully capable to handle most of the work by itself. This is supported from another comment on the following use cases that they can be done locally in the AV itself: maintain a steady speed at low RPM (UC15), shift-up early for information on improper shifting up tendencies of the vehicle (UC16), automatic tyre pressure check (UC17) and provide information about obstacles and blind spots on road (UC12).

We could reason that the division of functionality between the cloud and AV and the clear division of AV capabilities can play a major role in understanding which of the use cases can be relevant for cloud development or not. Different OEM's may develop autonomous trucks with different capabilities. This would vary the functionalities that fall under the control tower and perhaps this could be the reason where some of the existing literature's specify these use cases.

For the use case, running a complex battery lifetime model in the cloud (UC22), the comment received was that it requires much data transmission. This comment makes us to think that the respondent might have felt the use case helpful, but might also take into account of the possibility of implementing the use case. These might excite the team into discussion, but for the survey, such conflicts might possibly lead to neutral responses.

One of the comment received was that the use cases are known already and the respondent would find it of interest to learn some new use cases. This might be another reason for receiving some responses falling close to median. During the course of development of control tower, the team members might have come across use cases through literature study. But the purpose of conducting literature review close to being systematic is to help the team to understand all the functionalities covered by the control tower. The comment in fact establishes the truth of the literature review being effective and asserts the team to be in the right direction in control tower development.

The term "helpful" in the survey question being ambiguous in nature might have affected the results, due to varied interpretation by the respondents. The respondents were asked to rate how much do they agree that the use cases are helpful for the control tower for autonomous trucks. For the challenges it was asked helpful to be known for the control tower. But perhaps including these surveys after clearly explaining the study purpose in the presentation might have lead to a clear understanding among respondents while filling in the survey.

As seen in Figure 5.5, major portion of the responses for the challenges in implementation of control tower being helpful, appears positive. Responses being neutral or "I don't know" category were high, comparative to use cases. Challenges related to organizational policies, economic models, cloud membership, etc have received responses to be not knowing. The respondents being mostly developers might account for these responses for business and organizational challenges. The respondents during discussion had mentioned that they might require more details to understand the challenges completely. The same was echoed for few use cases too.

5.3.2 Final Survey

The final survey was held with the intention to measure the extent to which the findings solves the all the identified problem. Figure 5.6 shows the results of the survey gathered from 8 respondents. The values in y axis of the graph show the survey questions which are represented in clipped basic form to accommodate in the graph[3]. The results of the survey are discussed in problem solution pairs as in below implying the problem solving paradigm of DSR, with the problems corresponding to the sub causes in the Ishikawa diagram and the solution being the survey questions.

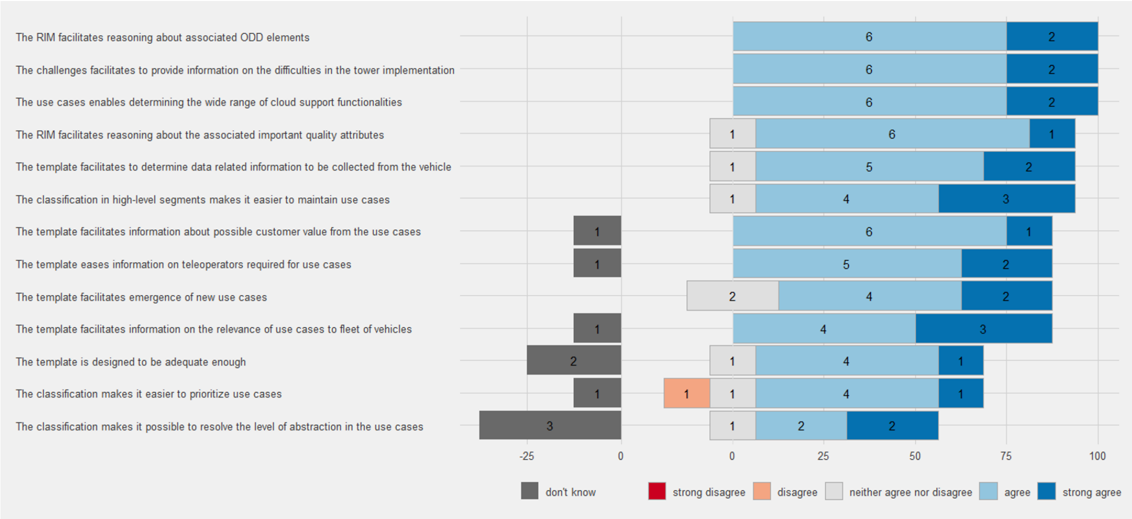

Figure 5.6: Results on the usefulness of the artifact in solving each of the identified problems

Problem. It is difficult to determine the cloud support functionalities that span over a wide range which makes it hard to provide useful use cases.

Solution. Identifying the potential list of use cases enables determining the cloud support functionalities that span over a wide range.

The results show that all the respondents agree or completely agree with the list of use cases. It is evident that since they were mined from existing literature's they were able to widely cover the functionalities.

Problem. It is difficult to resolve the level of abstraction among use cases which makes it challenging to clearly cluster use cases.

Solution. The classification makes it possible to resolve the level of abstraction in the use cases.

From the result, it is hard to agree that the classification resolves the level of abstraction, with 50% of the respondents with no clear opinion. So no conclusion could be drawn from it.

Problem. It is difficult to prioritize from the set of use cases.

[3] see Appendix A.4 to view the full questions.

Solution. The classification makes it easier to prioritize use cases.

We have a majority agreeing that the classification solves the problem regarding prioritization. However, we need to understand the reason behind disagreement. As one of the respondent comments:

"I think that the classification of the use cases should not hinder prioritization of any individual use case. On a high level it might work to decide on the focus for a specific project, but then it might hinder implementation of more important use cases in lower priority segment." – Interviewee B (Respondent)

Problem. It is hard to determine high level stable segments to maintain use cases.

Solution. The classification in high-level segments makes it easier to maintain use cases.

The results show that the segments in the classification successfully includes stable segments which can be relied upon as use cases increase in number. This is evident from the practical attempts to classify and strengthen with the support from literature.

Problem. Difficulties in control tower implementation are not known which makes it hard to provide useful use cases.

Solution. The list of challenges facilitates to provide information on the possible difficulties in the implementation of control tower.

All the respondents agree on the fact that the implementation difficulties are uncovered by the collected challenges.

Problem. No information on the applicability to fleet available which makes it hard to provide useful use cases.

Solution. The template through its element fleet applicability facilitates information on the relevance of use cases to fleet of vehicles.

The results show that the template element solves the problem.

Problem. New use cases might emerge continuously during the development.

Solution. The template through its element ODD elements facilitates emergence of new use cases.

From the results, it is clear that the template solves the problem by supporting all relevant use cases that might emerge during the process.

Problem. Information on teleoperators required is unknown.

Solution. The template through its element teleoperator requirement eases information on teleoperators required for use cases.

The results show that our solution solves the problem related to teleoperators.

Problem. There is lack of information on possible customer value of the use cases.

Solution. The template through its element expected benefit facilitates information about possible customer value from the use cases.

From the results we can agree that the element in the proposed template solves the problem of lack of information on imparted customer value from the use cases.

Problem. It is hard to determine the data requirements from use cases.

Solution. The template through its element data requirement facilitates to determine data related information to be collected from the vehicle.

The results show that the template is successful in capturing data requirements and thus makes it easier to determine.

Problem. It is challenging to design the artifact to be adequate enough.

Solution. By including only the concrete details determined through interview enables the artifact template to be adequate.

From the results, it is clear that half of the population agree with the solution solving the problem. But from the comment, it was evident to see the respondents looking for a way to utilize the template in determining quantifiable answers. As one of the respondents states:

"The template is a good start to get the right things into the use case, but the use case is only as good as the information it contains e.g. the Expected benefit should preferably be easy to quantify and be compared to the behaviour if the use case was not implemented. Maybe the template could help in formulating these benefits in a more measurable way." – Interviewee B (Respondent)

Problem. Difficult to realize associated ODD elements to the use case.

Solution. By connecting use cases and ODD elements, the information model facilitates reasoning about associated ODD elements.

From the results, it is evident that all respondents consider RIM to be successful in revealing ODD related information with the use case.

Problem. Difficult to realize associated quality attributes to the use cases.

Solution. By connecting use cases with quality attributes, the information model facilitates reasoning about the associated important quality attributes.

The results show that the reasoning of quality attribute is well established with the designed RIM. However, one of the comment received was:

"Is it really so that there is a mandatory relation between a use case and a Quality requirement? Couldn't there exist use case that are not connected to a Quality requirement or am I interpreting the Quality requirement wrong?" – Interviewee B (Respondent)

Perhaps, the survey including only the summarized findings might lack in conveying the model (and even other solutions) effectively, which might result in the respondents not completely deciding on the solution's usefulness.

5.3.3 Artifact Applicability

We evaluated whether the template can be applicable to the industrial use cases on control tower. It was found to be successfully applicable although requiring some modifications. The element *Existing Solutions* was found hard to fit in the template as the practitioners required detailed information of the solution (method used, open source software or not, evaluation result, etc.) depending on the source (research paper, software solution, etc.). This made to move the template element separately and descried rather in a paragraph or two. The template had an addition of a new element *Scenery*, which listed the area of operation of the truck, where the use case can be applicable. The scenery's were mainly parking area, confined area and public area.

6
Discussion

In this chapter we discuss the implications from the findings and shed light on some of the validity and ethical concerns.

6.1 Implications to Research

We see the designed artifact as a requirements tool-set as well as an initial list of relevant use cases and challenges, that can be utilized to support continuous development of a software. Use cases have a simplicity to represent a system's essence and have been found to drive not only requirements gathering but also the entire software development cycle. Several methodologies including the popular Rational Unified Process (RUP) are use case driven (Kulak et al., 2012). Yet the power of use cases are not realized in the continuous development process and this study has revealed the best practices to do so.

The list of use cases (see Section 4.2) provides an initial yet expandable set of use cases. We consider these to provide an overview to the new domain of control tower. The results from evaluation opens areas to define a "clear" role of control tower to lead to a consensus among OEMs and researchers. The classification (see Section 4.1) performed on a smaller set of use cases is clearly a solution to cluster set of use cases. As stated by Cockburn (2001): *"I was once sent over a hundred pages of use cases...below sea level... That requirements document did not serve either its writers or readers. The sender later sent me the six user-goal use cases that had replaced them, and said everyone found them easier to understand and work with"*. The use cases themselves were in high level of detail and were found in large numbers. The classification has ensured to cluster these based on the applications that the use cases support. A researcher would be interested to further investigate the support for architectural decisions through this classification and mapping of architectural units to the applications. Informed on the challenges (see Section 4.3) might be a contributing factor.

A major task of a researcher who constructs an artifact is the process of construction itself, with the main knowledge production happening (Crnkovic, 2010). The design thinking has helped to envisage the artifact that fills in the knowledge gap. The information model (see Section 4.4) bridges the connection between use cases, data requirements, ODD and quality attributes. It is unique to describe requirements for a new unexplored domain of automated driving and its cloud support.

Though there exists numerous use case templates, the constructed template (see Section 4.5) is unique to the domain. The existing templates allow support for filling in low level implementation details and the designed template in our study is a unique to be constructed at the higher level. The elements for the template have been selected based on the real problems identified after interviewing the practitioners themselves.

Use cases help to assure from a stakeholder's perspective that the system will have the required functionality and the quality attributes. A developer is forced to identify and include important elements in each use case from the predefined structure of a template (Kulak et al., 2012). However, the definition of "complete" after documenting a use case in the template is a subject of discussion. One way could be to determine if there are any missing steps in the implementation of use case based on the system or domain knowledge. A complete use case will include one main or basic flow and various alternate flows (the template element *Steps).* The other template elements are specific to the company needs which makes them optional to have. But in order to support the continuous development practice, some of the elements can be important to define documentation to be complete. For example, the element *ODD element*, to help for the elicitation of new use cases and the elements *Expected Benefit* and *Existing Solutions* to bridge the connection between managers and developers to speedup decisions. A validation for the documentation in the use case template to be said complete, can further be an interesting area to investigate.

6.2 Implications to Practitioners

The evaluation results confirmed that the artifact was applicable in the industry and was useful to address the problems discussed in Section 5.1.

The information model was designed at the end of the study iteration and is still in a prototype version to be fully relied upon by the practitioners. However, artifacts constructed in DSR are rarely full-grown information systems that are used in practice. Instead, artifacts are innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, and use of information systems can be effectively and efficiently accomplished (Denning, 1997). The results from the evaluation carried out with the practitioners assures the information model to be effective and communicates the necessary knowledge. In future, there is further room for the model to evolve and be fully functional.

From the search, hardly any existing support to manage requirements for continuous development practice was found. One of the tool in DevOps practice allows to create, automate, manage, analyze, and report requirements directly from the project[1]. The information model reveals important information of identification of new use cases using ODD elements. Practically, this ideation is the contribution to

[1]Requirements Management Tools built for Azure DevOps

continuous development by having new features available on a continuous basis.

The value of the template is that it is designed to connect the developers and the higher management. The information in the template mainly the *Expected benefit* and the *Segment* allows to take decisions whether the use case should be considered for the upcoming iteration of the continuous development cycle or not. The developers through the elements *Existing solution* and *Steps* can help to estimate the feature completion time from analysing the complexity of the given use case. These features of the template saves time and reduces communication gap by allowing to bring the developers and managers under a common platform to discuss and plan the road-map for the development.

The application of use case template with the industrial use cases, has also proved to be reliable. As one of the survey respondent states: *"...Maybe the template could help in formulating...in a more measurable way"*, the elements can be utilized to model and estimate the value of each use case which can guide practitioners in taking implementation decision. Conclusively, the artifact can be used by wide range of audiences in the company, mainly the developer, manager and architect. For example, in US5, the architect is enabled to take informed architectural decisions on gaining information of fleet related use cases; in UC7 the project manager can take effective decisions from knowing the value provided by the use cases to the customer, etc. Practically, the template is to guide the development and enable to make discussions and come with decisions.

6.3 Validity and Ethical Considerations

As stated by Runeson et al. (2008), we discuss the validity threats of this study in terms of internal validity, construct validity, external validity and reliability. Along with it, the study's conclusion validity are discussed. This section also sheds light on some of the ethical concerns and how the study has tried to adhere to the ethics.

Internal validity

Internal validity is concerned with confounding factors influencing the relationship between variables, treatment and results obtained. The use of interviews initially in problem identification allowed to explore the topic without being restricted to variables from the start. Before ending the interviews, the participants were asked for closing remarks which was helpful to expand the view. Continuous discussion with the supervisors and presenting and discussing the results helped to eliminate the researcher bias.

Construct validity

Construct validity concerned with the extent to which the study measures what it claims to be measuring. This is potentially compromised with clarifying the terms that are used in the study. To ensure non ambiguity with the definition of a control tower, apart from asking their opinion on the view of control tower, a video of a remote control tower of Saab was presented. The participants were asked about their opinion and how do they think the concept of control tower would vary when it comes to autonomous trucks. The terms used in the information model were refined from the continuous feedback received from supervisors. The elements used in the template are provided with definition in the Artifact section. The elements were also discussed during team presentations and are refined accordingly. The usage of DSR by itself enabled the researcher to follow the research questions and research on what was intended.

External validity

External validity is concerned with how generalized the findings are. Using design science study, the problem is learnt in a particular context and solved in that context, which makes generalizability as not a main goal. However, certain measures were taken. The survey respondents were asked to view the use case applicable to autonomous trucks in general and not specific to the company. Including the respondents with wide range of experience working in the similar domain also helped with it. For the classification and development of template and information model, use cases from existing literature's are used which makes the findings more generalized and applicable to other OEMs.

Reliability

Reliability is concerned with producing the same consistent result with other researchers on repeating the study. We have tried to ensure reliability by including the details of interview participants and survey respondents in terms of their role and years of experience. The interview template and survey questions have also been presented in the appendix. For the literature review, the databases and keywords used in the search process is also provided.

Conclusion validity

A conclusion validity measures how reasonable the conclusion drawn from the research is. It is concerned with the researcher establishing incorrect relationships from the findings. This involves concluding a relationship when there isn't any or missing a relationship. In the interview, the questions were open-ended, which allowed the participants to respond freely without subjected to pressure. They were assured of preserving confidentiality of information and promising to delete the recorded interview data before starting the interview. The thematic analysis of the non-confidential interview data and discussion of this with the supervisor helped to minimize the issues and establish correct findings.

For the survey, potential threats are that questions might not have been presented in the right order or it being too long to complete. In the initial survey, the questions were ordered under segments and in final survey, the questions were clustered according to each of the findings. Instead of one long survey, splitting them into 2 sessions during different days kept the length of the survey reasonable. We believe that by correctly ordering the questions, framing the surveys to be of reasonable length and by including "I don't know" responses helped to arrive at correct conclusion.

As of ethical concerns, for both the interview and survey, the participation was voluntary. An invitation was sent at least a week ahead for the anticipated participants. Further concerns are:

Informed consent

The interview participants were informed about the study purpose and their consent was taken orally for recording the interview. They were also informed that the recording will be deleted after the completion of this work. The survey respondents were informed about the motive to undertake the survey and the duration to complete it. The email invitees of the survey were also informed about the deadline to complete it. For both interview and survey, the participants were ensured of safeguarding anonymity and were informed on how their feedback will be used in this work.

Confidentiality and Anonymity

Anonymity requires the researcher not to know who the participants are and confidentiality involves the researcher knowing the identity, but not disclosing the information. Before the start of this study, confidentiality form was signed in agreement with the company as to not disclose any company related information. Both the interview and survey participants were assured of confidentiality. The industrial use cases employed to check the applicability of the template are kept confidential.

7 Conclusion and Outlook

Based on design science research, this study in the context of an automotive company has iteratively studied the problems emerged during the continuous development of the remote support solution (i.e control tower) for autonomous trucks and has designed solution which are validated through rigorous evaluation.

The list of use cases (see Section 4.2) has enabled to determine the wide range of control tower functionalities. Using use case classification (see Section 4.1) the maintenance of use cases are made easier through determining high level stable segments. Even prioritization of use cases is made easier. The collected list of challenges (see Section 4.3) reveals information on difficulties in the control tower implementation to take informed decisions. The use case template (see Section 4.5) facilitates information about fleet applicability, requirement of teleoperators, customer value and data requirements of individual use cases. The study realizes strong association between use cases and ODD elements which creates way to elicit new use cases. The requirements information model in this study (see Figure 4.1) realizes this association along with the association of use cases to quality attributes and data requirements.

The study can be evolved to work on further iterations to stabilize the developed information model to be applicable on a larger scale in different automotive companies. The study results can be refined to narrow the use case list to not include the ones also applicable to non-autonomous trucks. Future studies may also include use cases when there is a hierarchy of control towers and understanding how these use cases might vary or trigger based on different operational contexts. Studies in requirements engineering on affect of human factors can also be an interesting area to look into.

Bibliography

Adhikaree, A. et al. (2017). “Cloud-based battery condition monitoring platform for large-scale lithium-ion battery energy storage systems using internet-of-things (IoT).” In: *2017 IEEE Energy Conversion Congress and Exposition, ECCE 2017*. Vol. 2017-January. Department of Electrical Engineering and Computer Science, Texas AandM University-Kingsville, pp. 1004–1009. URL: https://doi.org/10.1109/ECCE.2017.8095896.

Aken, Joan van (Feb. 2001). “Management research based on the paradigm of the design sciences: The quest for field-tested and grounded technological Rules”. In: *Journal of Management Studies* 41. URL: https://doi.org/10.1111/joms.2004.41.issue-2.

Anda, Bente, Kai Hansen, and Gunhild Sand (2009). “An investigation of use case quality in a large safety-critical software development project.” In: *Information and Software Technology* 51.12, pp. 1699–1711. URL: https://doi.org/10.1016/j.infsof.2009.04.005.

Aydin, M., M. Samarah, and K.O. Elish (2018). “Enabling smart cities through v2x communication.” In: *2018 IEEE 4th International Conference on Computer and Communications, ICCC 2018*. Department of Computer Science, Florida Polytechnic University, pp. 550–554. URL: https://doi.org/10.1109/CompComm.2018.8780839.

Baskerville, Richard et al. (2018). “Design Science Research Contributions: Finding a Balance between Artifact and Theory.” In: *Journal of the Association for Information Systems* 19.5, pp. 358–376. URL: https://doi.org/10.17705/1jais.00495.

Berander, Patrik and Anneliese Andrews (2005). “Requirements Prioritization”. In: *Engineering and Managing Software Requirements*. Ed. by Aybüke Aurum and Claes Wohlin. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 69–94. URL: https://doi.org/10.1007/3-540-28244-0_4.

Bitam, Salim, Abdelhamid Mellouk, and Sherali Zeadally (2015). “VANET-cloud: a generic cloud computing model for vehicular Ad Hoc networks”. In: *IEEE Wireless Communications* 22.1, pp. 96–102.

Bosch, Jan (2014). “Continuous software engineering: An introduction”. In: *Continuous software engineering*. Springer, pp. 3–13. URL: https://doi.org/10.1007/978-3-319-11283-1.

Braun, Virginia and Victoria Clarke (2006). “Using thematic analysis in psychology”. In: *Qualitative Research in Psychology* 3.2, pp. 77–101. URL: https://www.tandfonline.com/doi/abs/10.1191/1478088706qp063oa.

Bsaybes, Sahar, Alain Quilliot, and Annegret K. Wagler (2019). “Fleet management for autonomous vehicles using flows in time-expanded networks.” In: *TOP* 2, pp. 288–311. URL: https://doi.org/10.1007/s11750-019-00506-4.

Burton, Simon et al. (2020). “Mind the gaps: Assuring the safety of autonomous systems from an engineering, ethical, and legal perspective.” In: *Artificial Intelligence* 279. ISSN: 0004-3702. URL: https://doi.org/10.1016/j.artint.2019.103201.

Cockburn, Alistair (2001). *Writing effective use cases.* The Agile software development series. Addison-Wesley.

Cordeil, Maxime, Tim Dwyer, and Christophe Hurter (2016). “Immersive Solutions for Future Air Traffic Control and Management”. In: *Proceedings of the 2016 ACM Companion on Interactive Surfaces and Spaces*. New York, NY, USA: Association for Computing Machinery, pp. 25–31. ISBN: 9781450345309. URL: https://doi.org/10.1145/3009939.3009944.

Crnkovic, Gordana Dodig (2010). “Constructive research and info-computational knowledge generation”. In: *Model-Based Reasoning in Science and Technology.* Springer, pp. 359–380.

Czarnecki, Krzysztof (July 2018a). *Operational Design Domain for Automated Driving Systems - Taxonomy of Basic Terms*. URL: https://doi.org/10.13140/RG.2.2.18037.88803.

– (July 2018b). *Operational World Model Ontology for Automated Driving Systems - Part 1: Road Structure*. URL: https://doi.org/10.13140/RG.2.2.15521.30568.

– (July 2018c). *Operational World Model Ontology for Automated Driving Systems - Part 2: Road Users, Animals, Other Obstacles, and Environmental Conditions.* URL: https://doi.org/10.13140/RG.2.2.11327.00165.

Daw, Andrew, Robert C. Hampshire, and Jamol Pender (2019). “Beyond Safety Drivers: Staffing a Teleoperations System for Autonomous Vehicles”. In: *ArXiv* abs/1907.12650.

Denning, Peter J. (Feb. 1997). "A New Social Contract for Research". In: *Commun. ACM* 40, pp. 132–134. ISSN: 0001-0782. URL: https://doi.org/10.1145/253671.253755.

Dresch, Aline, Daniel Pacheco Lacerda, and José Antônio Valle Antunes Jr (2015). *Design Science Research: A Method for Science and Technology Advancement.* Springer International Publishing. ISBN: 9783319073743. URL: 10.1007/978-3-319-07374-3.

Glinz, Martin (2007). "On Non-Functional Requirements". In: *15th IEEE International Requirements Engineering Conference (RE 2007)*, pp. 21–26.

Herrnleben, S. (1) et al. (2019). "Towards Adaptive Car-to-Cloud Communication." In: *2019 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2019.* (1)Universität Würzburg, Department of Software Engineering, pp. 119–124. URL: https://doi.org/10.1109/PERCOMW.2019.8730766.

Hevner, Alan and Samir Chatterjee (2010). "On Design Theory". In: *Design Research in Information Systems: Theory and Practice.* Boston, MA: Springer US. URL: https://doi.org/10.1007/978-1-4419-5653-8_4.

Hobert, Laurens et al. (2015). "Enhancements of V2X Communication in Support of Cooperative Autonomous Driving." In: *IEEE COMMUNICATIONS MAGAZINE* 53.12, pp. 64–+. URL: https://doi.org/10.1109/mcom.2015.7355568.

Ishikawa, Kaoru (1990). *Introduction to quality control.* Productivity Press.

ISO/IEC 25010 (2011). *ISO/IEC 25010:2011, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.*

ITRL, Frank Giang-Developer-KTH and Konrad Tollmar (2018). "Project Report Automated Vehicles Traffic Control Tower (AVTCT)". In:

Jacobson, Ivar (1993). *Object-oriented software engineering : a use case driven approach.* Pearson Education India.

Kasauli, Rashidah et al. (Feb. 2017). "Adding Value Every Sprint: A Case Study on Large-Scale Continuous Requirements Engineering". In:

Kauppi, Arvid et al. (July 2017). "Future Train Traffic Control: Control by Replanning: Supporting the Integrated Railway". In: pp. 296–305. ISBN: 9781315089201. URL: https://doi.org/10.4324/9781315089201-27.

King, Nigel (2004). “Using templates in the thematic analysis of text”. In: *Essential guide to qualitative methods in organizational research*, p. 256. URL: https://doi.org/10.4135/9781446280119.n21.

Kitchenham, B. and S Charters (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering.*

Knauss, Alessia and Daniela Damian (2015). “The Capture and Evolution of Contextual Requirements: The Case of Adaptive Systems”. PhD thesis.

Knauss, Alessia, Daniela Damian, et al. (2016). “ACon: A learning-based approach to deal with uncertainty in contextual requirements at runtime”. In: *Information and Software Technology* 70, pp. 85–99. ISSN: 0950-5849. URL: https://doi.org/10.1016/j.infsof.2015.10.001.

Krusche, Stephan and Lukas Alperowitz (2014). “Introduction of Continuous Delivery in Multi-Customer Project Courses”. In: *Companion Proceedings of the 36th International Conference on Software Engineering.* New York, NY, USA: Association for Computing Machinery, pp. 335–343. ISBN: 9781450327688. URL: https://doi.org/10.1145/2591062.2591163.

Kulak, Daryl and Eamonn Guiney (2012). *Use cases: requirements in context.* Addison-Wesley.

Kumar, Swarun, Shyamnath Gollakota, and Dina Katabi (Aug. 2012). “A cloud-assisted design for autonomous driving”. In: *MCC’12 - Proceedings of the 1st ACM Mobile Cloud Computing Workshop.* URL: https://doi.org/10.1145/2342509.2342519.

Kvale, Steinar (1996). *Interviews: an introduction to qualitative research interviewing.* Sage Publications.

Likert, Rensis (1932). “A technique for the measurement of attitudes”. In: *Archives of Psychology* 22, pp. 5–55.

Liu, Shaoshan et al. (2017). “Implementing a Cloud Platform for Autonomous Driving.” In: *arXiv preprint arXiv:1704.02696.*

Liu, Zheng et al. (2016). “Unmanned surface vehicles: An overview of developments and challenges”. In: *Annu. Rev. Control.* 41, pp. 71–93.

Lovas, R. (1) et al. (2019). “PaaS-oriented IoT platform with connected cars use cases.” In: *Proceedings - 2018 International Conference on Sensor Networks and Signal Processing, SNSP 2018.* (1)Laboratory of Parallel, Distributed Systems, Institute for Computer Science, and Control, Hungarian Academy of Sciences

(MTA SZTAKI), pp. 409–420. URL: https://doi.org/10.1109/SNSP.2018.00085.

Maheswaran, Muthucumaru, Tianzi Yang, and Salman Memon (2019). "A Fog Computing Framework for Autonomous Driving Assist: Architecture, Experiments, and Challenges." In: *arXiv preprint arXiv:1907.09454*.

Marosi, A.C. (1) et al. (2018). "A novel IoT platform for the era of connected cars." In: *2018 IEEE International Conference on Future IoT Technologies, Future IoT 2018*. (1)Laboratory of Parallel, Distributed Systems, Institute for Computer Science, and Control, Hungarian Academy of Sciences, pp. 1–11. URL: https://doi.org/10.1109/FIOT.2018.8325597.

Milani, F. (1) and C. (2) Beidl (2019). "Cloud-based Vehicle Functions: Motivation, Use-cases and Classification." In: *IEEE Vehicular Networking Conference, VNC*. (1)Corporate Sector Research and Advance Engineering, Robert Bosch GmbH. URL: https://doi.org/10.1109/VNC.2018.8628342.

Moisiadis, F. (2000). "Prioritising use cases and scenarios." In: *Proceedings 37th International Conference on Technology of Object-Oriented Languages Systems TOOLS-Pacific 2000*, p. 108.

Najada, Hamzah Al and Imad Mahgoub (2016). "Autonomous vehicles safe-optimal trajectory selection based on big data analysis and predefined user preferences." In: *2016 IEEE 7th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*, pp. 1–6. ISSN: 9781509014965. URL: https://doi.org/10.1109/UEMCON.2016.7777922.

Niu, N. et al. (2018). "Requirements Engineering and Continuous Deployment". In: *IEEE Software* 35.2, pp. 86–90.

Ozatay, Engin et al. (2014). "Cloud-Based Velocity Profile Optimization for Everyday Driving: A Dynamic-Programming-Based Solution." In: *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 15.6, pp. 2491–2505. URL: https://doi.org/10.1109/TITS.2014.2319812.

Palinkas, Lawrence A et al. (2015). "Purposeful sampling for qualitative data collection and analysis in mixed method implementation research". In: *Administration and policy in mental health and mental health services research* 42.5, pp. 533–544.

Pfleeger, Shari Lawrence and Barbara A. Kitchenham (Nov. 2001). "Principles of Survey Research: Part 1: Turning Lemons into Lemonade". In: *SIGSOFT Softw. Eng. Notes* 26.6, pp. 16–18. ISSN: 0163-5948. URL: https://doi.org/10.1145/505532.505535.

Rai, Arun (June 2017). “Editor’s Comments: Avoiding Type III Errors: Formulating IS Research Problems That Matter”. In: *MIS Q.* 41.2, pp. iii–vii.

Runeson, Per and Martin Höst (Dec. 2008). “Guidelines for conducting and reporting case study research in software engineering”. In: *Empirical Software Engineering* 14, p. 131. ISSN: 1573-7616. URL: https://doi.org/10.1007/s10664-008-9102-8.

Rupp, J.D. and A.G. King (2010). “Autonomous driving - A practical roadmap.” In: *SAE Technical Papers*. URL: https://doi.org/10.4271/2010-01-2335.

Saldaña, Johnny (2015). *The coding manual for qualitative researchers*. Sage.

Savor, Tony et al. (2016). “Continuous deployment at Facebook and OANDA”. In: *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, pp. 21–30.

Schmidt, M. et al. (2009). “Remote airport traffic control center with augmented vision video panorama.” In: *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*. German Aerospace Center (DLR), Institute of Flight Guidance, 4.E.21–4.E.215. URL: https://doi.org/10.1109/DASC.2009.5347479.

Shahin, Mojtaba et al. (2017). “Beyond continuous delivery: an empirical investigation of continuous deployment challenges”. In: *2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, pp. 111–120.

Stephan, Olariu, Eltoweissy Mohamed, and Younis Mohamed (2011). “Towards autonomous vehicular clouds.” In: *EAI Endorsed Transactions on Mobile Communications and Applications* 1, p. 1. URL: https://doi.org/10.4108/icst.trans.mca.2011.e2.

Tiwari, Saurabh and Atul Gupta (2017). “Investigating comprehension and learnability aspects of use cases for software specification problems.” In: *Information and Software Technology* 91, pp. 22–43. URL: https://doi.org/10.1016/j.infsof.2017.06.003.

Vagia, M. (1) and E.J. (2) Rødseth (2019). “A taxonomy for autonomous vehicles for different transportation modes.” In: *Journal of Physics: Conference Series*. Vol. 1357. 1. (1)SINTEF Digital. URL: https://doi.org/10.1088/1742-6596/1357/1/012022.

Wachenfeld, W. (1) et al. (2016). *Use cases for autonomous driving*. (1)Institute of Automotive Engineering - FZD, Technische Universität Darmstadt: Springer Berlin Heidelberg. URL: https://doi.org/10.1007/978-3-662-48847-8_2.

Weiss, Robert Stuart. (1995). *Learning from strangers: the art and method of qualitative interview studies.* The Free Press.

Whaiduzzaman, Md et al. (2014). "A survey on vehicular cloud computing". In: *Journal of Network and Computer Applications* 40, pp. 325–344. URL: https://doi.org/10.1016/j.jnca.2013.08.004.

Wiegers, Karl Eugene (1999). *Software Requirements.* USA: Microsoft Press.

Wohlrab, Rebekka, Eric Knauss, and Patrizio Pelliccione (2020). "Why and how to balance alignment and diversity of requirements engineering practices in automotive". In: *Journal of Systems and Software* 162, p. 110516. ISSN: 0164-1212. URL: https://doi.org/10.1016/j.jss.2019.110516.

Yuh, J., Giacomo Marani, and D. Blidberg (Oct. 2011). "Applications of marine robotic vehicles". In: *Intelligent Service Robotics* 4, pp. 221–231. URL: https://doi.org/10.1007/s11370-011-0096-5.

Zehe, Alexandra Katharina and Frank-Martin Belz (2016). "Analyzing Complex Relationships in Organizational Research by Means of Video Elicitation Interviews." In: *Academy of Management Annual Meeting Proceedings* 2016.1, p. 1. URL: https://doi.org/10.5465/ambpp.2016.16115abstract.

Zhao, Xiaoyun, Rami Darwish, and Anna Pernestål (2019). "Automated vehicle traffic control tower - the bridge to next level automation". In:

A
Appendix 1

A.1 Interview Template

The provided interview instrument was used during the first round of iteration on problem investigation. During this first round 9 interviews were conducted with 9 interviewees. The role of these interviewees and their assigned task contribution are listed in Table 3.1. The interview questions were based on the use cases, control tower and the ODD components, to get a clearer picture of what the problem is about and what are the expectations from the company.

Table A.1: Interview questions

	General questions on the interviewee
1	What is your role in the company?
2	In which team do you work?
	• Would you briefly describe your team tasks?
	Questions on the control tower
3	According to your view point, what is a control tower for autonomous trucks?
4	How is your team contributing to the control tower?
5	Where is Volvo Autonomous Solutions right now, in its advancements of the control tower?
6	Why do autonomous trucks need a control tower?
7	If we look at level 4 autonomy, the truck can run in autonomous mode only under limited conditions.
	• In this case, what could be the role of the control tower?
8	For level 5 autonomy, the trucks do not require human attention and it has the ability to drive autonomously in all conditions. In this case, will there be any role of control tower?
	• If Yes, how?
	• If No, how?
9	Now that we know how beneficial a control tower could be for autonomous trucks, what could be some of the challenges associated with it?

10	(Based on video[1]) What is your picture of the functioning of the control tower for autonomous trucks?
	Questions on the ODD components
11	We know that ODD components especially the road conditions might be critical for the vehicle when on drive. For example, slippery road, damaged road surface, etc. But is there any role of ODD components in relation to the control tower ? • If Yes, why? • If No, why?
	Questions on use cases
12	What makes a good use case for the control tower according to your view?
13	If need to be prioritized, which aspects can be critical?
	Wrap up question
14	Is there anything that you would like to add about the use cases, ODD components or the control tower?

A.2 Draft Use Case Template and Example

Table A.2: Use case template for control tower solution

Use case	<number> <the name should be the goal as a short active verb phrase>
Segment	<name of the segment to which it belongs>
Summary	<a longer description of the goal>
Aspect	<one word list of areas the use case can be helpful>
Expected benefit	<list of all the possible benefits to the AV and the external parties>
Fleet applicability	<Boolean>
Human operator required	<Boolean>
Data transfer	<a short statement on immediate or delayed data transfer>
Type of data	<list of data required>

Display/alert principle	<the action that starts the use case>
Description	Step Action 1 <steps of the scenario from trigger to goal delivery, and any cleanup after> 2 <...>
Challenges	<what might cause the use case to fail>
Existing solutions	<list of software's that already implement the use case>
ODD elements	<a hierarchy of all the ODD's from the list of taxonomy that impacts the use case>

Table A.3: Use case template example

Use case1	HD map update
Context	AV detects change in its HD map
Summary	AV uses its HD map to navigate safely during its operation. It can detect any changes in its rout. If detected, it updates its map and triggers the cloud. The cloud can analyze the change and update its own HD map. The change detected by the AV can be notified by the cloud to the external authorities or can be notified to other AV's along the same route.
Aspect	Social, operation efficiency, safety
Expected benefit	• The authorities can be updated about any damage in the roadside structures and help in maintenance • AV's can get access up-to-date maps with the up-to-date traffic database
Fleet applicability	Yes
Human operator required	No
Data transfer	Real time
Type of data	GPS, world model object data
Display/alert principle	Trigger by the AV to update after analyzing change

Description	Step Action 1 Cloud receives notification to update its HD map 2 Updates HD map 3 Notifies other AV's 4 Nofities road authorities
Challenges	In communication layer: varying latency and delay time of network
Existing solutions	• NVIDIA's DRIVE Mapstream and DRIVE Mapservices • AND
ODD elements	Figure A.1

The possible cloud based applications by knowing the ODD elements visualized in Figure A.1 could be: Traffic sign recognition data collection use case from the element traffic sign. The collected data can keep track of the condition of these road side structures and can even report missing cases. Another use case could be to provide information about obstacles and blind spots to optimize the trajectory planning. Temporary road structures like barricades indicate temporary traffic block or diversion. Knowing information early can be used to re route other AVs that follow the same route. School crossings are supervised by school crossing guards during specified hours and during regular school periods (Czarnecki, 2018c). Analysis of this data, can be a solution to the use case on autonomous mitigation of recurring congestion.

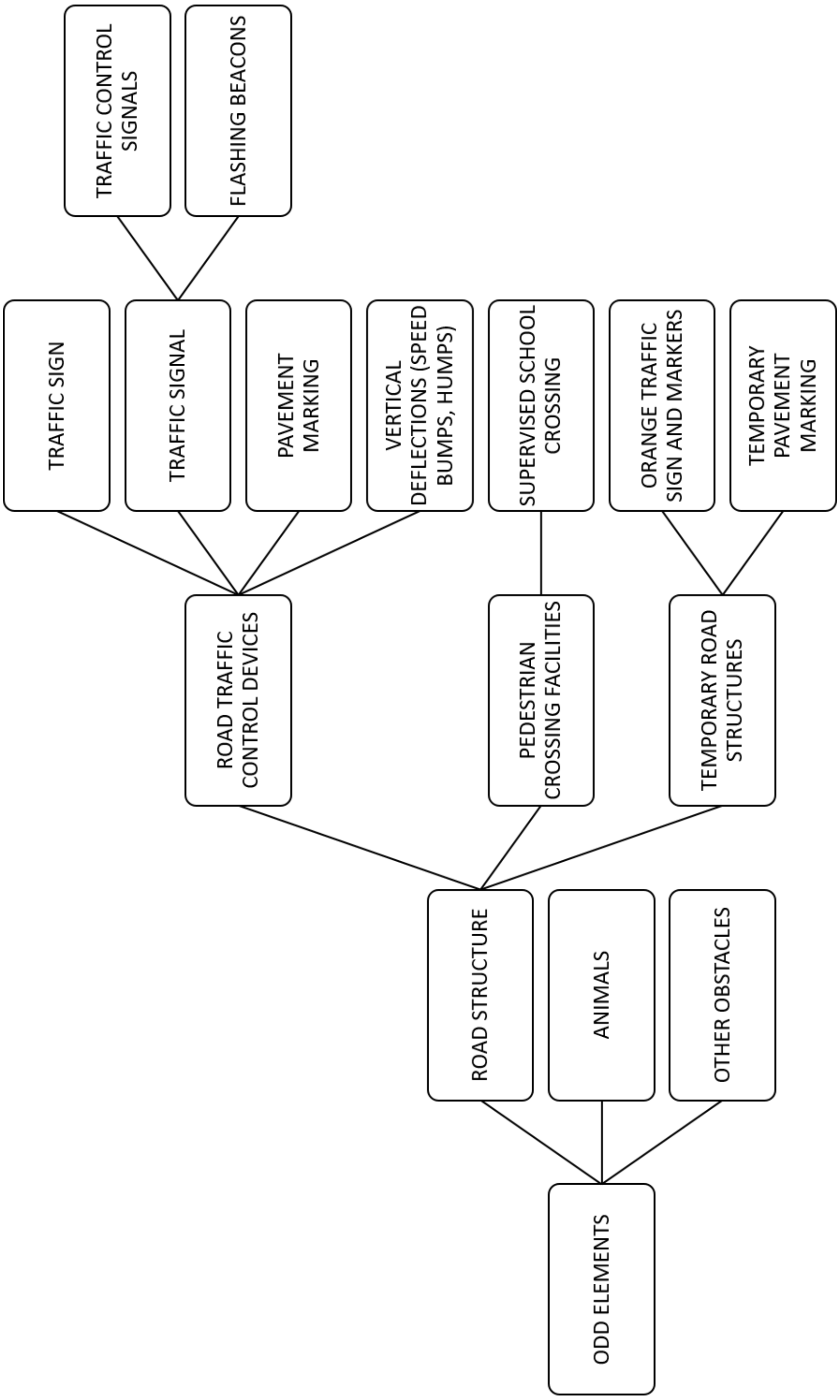

Figure A.1: ODD elements for template example

A.3 Initial Survey Template

soSci
oFb - der onlineFragebogen

0% completed

Welcome to this survey!

1. What is your role in the company? (Ex: manager, developer, architect, etc.)

2. What are your years of experience?

Next

Supriya – 2020

3. How much do you agree that the following use cases are helpful for the control tower of autonomous trucks?

The following use cases belong to the segment Improved Traffic Performance.

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
U1: Synchronizing traffic lights after clearing an accident						
The computational resources of all the vehicles will be used to recommend to a higher authority a way of rescheduling the traffic lights that will serve the purpose of de-congesting the afflicted area as fast as possible	○	○	○	○	○	○
U2: Autonomous mitigation of recurring congestion						
The vehicles in the vicinity will be able to query the plan of each other at time of rerouting during congestion and estimate the impact on local roads to prevent vehicle flooding	○	○	○	○	○	○
U3: Efficient use of road space						
Implement allocation algorithms for efficiently dividing road space among competing vehicles. The lane configurations are customized to safely pack the increasingly heterogeneous traffic onto the road space.	○	○	○	○	○	○
U4: Optimizing traffic signals						
The vehicular cloud is sensing the segment traffic congestion by transferring the time, GPS coordinates and final destination to a navigation server through onboard vehicle navigators. The navigation server implemented in the cloud is in charge of computing the optimal routes by constructing traffic load map and the traffic pattern matrix and estimating road segment loads and delays.	○	○	○	○	○	○
U5: Dynamic management of high occupancy vehicle (HOV) lanes						
Factoring data from sensors on board of the individual vehicles, e.g. occupancy sensors, and local traffic intensity in order to optimally configure the HOV lanes	○	○	○	○	○	○

Any comments?

4. How much do you agree that the following use cases are helpful for the control tower of autonomous trucks?

The following use cases belong to the segment Improved Traffic Safety.

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
U1: Information sharing through an outside-the-car observer The observer can pre-process the data feeds obtained from the vehicles or other road objects before passing them along to other vehicles. For instance, pothole information or changing driveability conditions during wildfires, etc.	○	○	○	○	○	○
U2: Provide information about obstacles on road A cloud-assisted system for autonomous driving can significantly improve the safety of autonomous driving, by providing vehicles greater access to critical information and upcoming barriers	○	○	○	○	○	○

Any comments?

5. How much do you agree that the following use cases are helpful for the control tower of autonomous trucks?

The following use cases belong to the segment Reduced Local Emissions.

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
U1: Maintain a steady speed at low RPM Intensive braking and unnecessary speeding both wastes fuel and increases safety risks. The analysis can help the vehicle with acceleration, etc.	○	○	○	○	○	○
U2: Shift-up early for information on improper shifting up tendencies of the vehicle With the help of the vehicles RPM and speed information improper shifting up tendencies and habits can be examined and corrected later on	○	○	○	○	○	○
U3: Automatic Tyre pressure check before drive, after and at high speeds Automatic checks can be performed frequently at least once a month and before driving, and at high speeds. Fuel saving can be estimated from this value	○	○	○	○	○	○
U4: Cloudbased algorithm that generates an optimal speed trajectory to reduce the fuel consumption The server generates a route to reach intended destination, collects the associated traffic and geographical information, and solves the optimization problem by a spatial domain dynamic programming (DP) algorithm that utilizes accurate vehicle and fuel consumption models to determine the optimal speed trajectory along the route	○	○	○	○	○	○

Any comments?

6. How much do you agree that the following use cases are helpful for the control tower of autonomous trucks?

The following use cases belong to the segment Increased Individual Needs.

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
U1: Managing evacuation The information collected through individual vehicles can be integrated together with information about traffic on a road available through inductive loops, cameras, roadside sensors and surveys to form a real time picture of the road situation	○	○	○	○	○	○
U2: Detecting dead animals and other objects on the road Cleaning crew and police can get notified in real-time where exactly these objects lie. This can allow them to remove the objects very quickly from the road	○	○	○	○	○	○
U3: Dynamic management of parking facilities By real-time pooling the information from the vehicles in the vicinity about the availability of parking at various locations inside the city, vehicles can be directed to the most promising location where parking is (still) available	○	○	○	○	○	○
U4: Charging station managements Cloud connectivity provides fine-grained live data to optimize arrival and battery estimations and for using the vehicles as IoT sensors. Example of an optimization is the allocation of charging stations by booking an estimated time slot	○	○	○	○	○	○

Any comments?

7. How much do you agree that the following use cases are helpful for the control tower of autonomous trucks?

The following use cases belong to the segment Enhanced Data Collection.

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
U1: Weather report conditions and forecast Weather nowcasting shows very short-term weather predictions based on recent and localized measurements. Incorporating data, collected from a large set of vehicles allows these predictions to be more precise and detailed than traditional weather forecasting	○	○	○	○	○	○
U2: Traffic sign recognition data collection A statistical analysis on comparisons between existing traffic sign database and the one collected from vehicle can be used to identify new, changed, missing/stolen traffic signs and update the traffic sign database with the respective information. This database can provide useful information not only for the traveling users, but can also provide the road maintenance services with notifications where to check the road signs for dirt or if they were stolen	○	○	○	○	○	○
U3: Monitoring of accidents Vehicles with their array of sensors can perform as clients by collecting field data related to accidents. The spatial (map data) and non-spatial (sensor parameters) data is sent to a database center where it gets merged into one database. The main system analyses the incoming data to achieve various types of statistical reports	○	○	○	○	○	○
U4: Backoffice providing details of all the errors, warning, etc. A cloud-based framework that serves as one of the user interfaces for accessing the different functionalities of the platform, both for users and administrators	○	○	○	○	○	○
U5: Road condition monitoring The collected vertical acceleration data from the vehicle allows identifying road faults and for assessing the vibration load. These vibration values combined with collected trajectory and speed data can be used to profile routes according to road quality, goods vibration load, etc.	○	○	○	○	○	○

8. How much do you agree that the following use cases are helpful for the control tower of autonomous trucks?
The following use cases belong to the segment Other Functionalities.

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
U1: Regular maintenance						
Collecting service information and recommending parts and service stations	○	○	○	○	○	○
U2: HD map generation						
As the autonomous vehicles are moving and collecting new LiDAR scans, they compare in real time the new LiDAR scans against the grid map with initial position estimates provided by GPS and/or Inertial Measurement Unit (IMU), which then assists these vehicles in precisely self-localizing in real-time	○	○	○	○	○	○
U3: Running a complex battery lifetime model in the cloud						
Makes it possible to predict the state of health of high voltage batteries	○	○	○	○	○	○
U4: Plausibility checks						
Manipulation in electronic control unit (ECU) software, sensor specifications or transmitted data can hamper the vehicle's normal operation. In order to verify the plausibility of sensors and ECU software, the models of vehicle software with series data and initial sensor specification are stored in the cloud and can be compared	○	○	○	○	○	○
U5: Remote engine start						
It allows to remotely start and run the vehicle for a definable period. This allows e.g., to control the interior temperature using the climate control system, to pre-warm the engine, etc.	○	○	○	○	○	○

Any comments?

9. How much do you agree that the following challenges are helpful to be known for the control tower of autonomous trucks?

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
C1: Varying latency and delay time of cellular network						
The transmission delay over the wireless link, delay induced by the security mechanisms (generation and verification of signature and certificate, respectively), etc	○	○	○	○	○	○
C2: Data load control						
Data load in the network gets amplified by high vehicle density corresponding to exchange in control message and data	○	○	○	○	○	○
C3: Cellular network availability						
It plays an important role in accessing the cloud based vehicle functions (CB-VF), e.g. in case of temporary lack of cellular coverage in a dead-spot zone, the on-board ECUs do not have access to the CB-VFs	○	○	○	○	○	○
C4: Highly reliable packet delivery						
A lost or erroneous message might cause a malfunction of the vehicle control algorithms and create a safety risk	○	○	○	○	○	○
C5: High message rate						
In the first generation of V2X communication systems (1G-V2X), vehicles periodically broadcast safety messages with an interval between 100 ms and 1 s. This will lead to high message rate as the monitored vehicles increase	○	○	○	○	○	○
C6: Elastic mobile architecture						
The architecture must be developed to accommodate changing application demands and resource availability on the move	○	○	○	○	○	○
C7: Robust architecture						
The architectural blocks must be designed to withstand structural stresses induced by the inherent instability in the operating environment	○	○	○	○	○	○

C8: Shared responsibility

At times for the ultimate vehicle drive actions it is best to have inputs from both cloud and vehicle. The distributed characteristic of vehicle applications between cloud and on-board ECUs requires an appropriate partitioning

C9: Managing highly dynamic cloud membership

There is a critical need to efficiently manage mobility, resource heterogeneity (including sensing, computation and communication), trust and vehicle membership (change in interest, change in location, resource denial and/or failure)

C10: Privacy and security challenges

Includes data security, cloud access control, securing vehicular communication, securing vehicular communication, securing location information such as traffic status reports, collision avoidance, etc.

C11: Trust assurance

Resolution of a problem such as on considering the workable schedule of the traffic lights produced by AV in a traffic jam, resides in some form of a trust relationship that needs to be forged between the municipal or country authority and the cloud

C12: Effective operational policies

They are needed for seamless inter-operation, decision support, establishing accountability metrics, standardization, regulations, and even local and national policy making

C13: Autonomous vehicular cloud (AVC) utility computing

Need for economic models and metrics to determine reasonable pricing and billing for AVC services

C14: Federation of different clouds

Interoperability of different types of clouds, connection, synchronization, and reliability and efficiency should be addressed

C15: Sensing and aggregation data

New research solutions are required to efficiently sense and aggregate various types of sensor data, including traffic data, vehicle's health information, information about the environment (disasters, fire, etc.), movements of vehicles and citizens on roads, etc.

Any final comments before the end of the survey?

soSci oFb - der onlineFragebogen

Thank you for completing this questionnaire!

We would like to thank you very much for helping us.

Your answers were transmitted, you may close the browser window or tab now.

Supriya – 2020

A.4 Final Survey Template

soSci
oFb - der onlineFragebogen

0% completed

Welcome to this survey!

The survey is intended to be a part for my thesis titled:

A REQUIREMENT ENGINEERING APPROACH FOR SUPPORTING CONTINUOUS DEVELOPMENT OF CLOUD BASED SUPPORT FOR AUTONOMOUS TRUCKS

The findings of this thesis are:
1. List of use cases
2. Use case classification
3. List of challenges
4. Use case template
5. Information model

The survey includes summary of the thesis findings followed by questions which measure the extent to which the findings solve the identified problems.

The survey might take around 15min to complete and the results will be anonymous.

1. Which company do you work in?

2. What is your role in the company? (Ex: manager, developer, architect, etc.)

3. What are your years of experience?

Next

Supriya, CHALMERS – 2020

4. How much do you agree with the following statements on control tower use cases, its classification and challenges associated with implementation of control tower?

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
Identifying the potential "list of use cases" enables determining the cloud support functionalities that span over a wide range	○	○	○	○	○	○
The "classification" makes it possible to resolve the level of abstraction in the use cases	○	○	○	○	○	○
The "classification" makes it easier to prioritize use cases	○	○	○	○	○	○
The "classification" in high-level segments makes it easier to maintain use cases	○	○	○	○	○	○
The "list of challenges" facilitates to provide information on the possible difficulties in the implementation of control tower	○	○	○	○	○	○

6. How much do you agree with the following statements on the use case template?

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
The "template" through its element fleet applicability facilitates information on the relevance of use cases to fleet of vehicles	○	○	○	○	○	○
The "template" through its element ODD elements facilitates emergence of new use cases	○	○	○	○	○	○
The "template" through its element teleoperator requirement eases information on teleoperators required for use cases	○	○	○	○	○	○
The "template" through its element expected benefit facilitates information about possible customer value from the use cases	○	○	○	○	○	○
The "template" through its element data requirement facilitates to determine data related information to be collected from the vehicle	○	○	○	○	○	○
By including only the concrete details determined through interview enables the artifact "template" to be adequate enough	○	○	○	○	○	○

8. How much do you agree with the following statements on the information model?

	strongly disagree	disagree	neither agree nor disagree	agree	strongly agree	don't know
By connecting use cases and ODD elements, the "information model" facilitates reasoning about associated ODD elements	○	○	○	○	○	○
By connecting use cases with quality attributes, the "information model" facilitates reasoning about the associated important quality attributes	○	○	○	○	○	○

9. Any final comments before the end of the survey?