Mediator

Mediator System and Functional Requirements

Deliverable D1.4 – WP1 – Public

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 814735.

Mediator System and Functional Requirements

Work package 1, Deliverable D1.4

Please refer to this report as follows:
Cleij, D., Bakker, B., Borowsky, A., Christoph, M., Fiorentino, A., van Grondelle, E., Mano, D, van Nes, N. (2020). Mediator System and Functional Requirements, Deliverable D1.4 of the H2020 project MEDIATOR.

Project details:	
Project start date: **Duration:** **Project name:**	01/05/2019 48 months MEDIATOR – MEdiating between Driver and Intelligent Automated Transport systems on Our Roads
Coordinator:	Prof. dr Nicole van Nes | SWOV – Institute for Road Safety Research Bezuidenhoutseweg 62, 2594 AW, The Hague, The Netherlands
	This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 814735.

Deliverable details:	
Version: **Dissemination level:** **Due date:** **Submission date:**	Final v2 Public (PU) 30/06/2021 30/06/2021 – resubmission after P1 review

Lead contractor for this deliverable:
Diane Cleij – SWOV Institute for Road Safety Research

Report Author(s):	Cleij, D., Christoph, M., van Nes, N.- SWOV Institute for Road Safety Research, Netherlands Bakker, B. - Cygnify, The Netherlands Borowsky, A. - Ben-Gurion University of the Negev, Israel Fiorentino, A. - Fiat Chrysler Automobiles, Italy van Grondelle, E. - Delft University of Technology, Netherlands Mano, D. – Veoneer, Sweden

Revision history

Date	Version	Reviewer	Description
03/12/2020	Preliminary draft	Michiel Christoph (WP leader), Ingrid van Schagen (QA officer) – SWOV Institute for Road Safety Research	Overall check before formal QA
07/12/2020	Draft for QA	Prof. Dr. Linda Ng Boyle- University of Washington	External review
07/12/2020	Draft for QA	Dr Matthias Beggiato - Chemnitz University of Technology	Internal review
21/12/2020	Final draft	Ingrid van Schagen (QA officer), Ernst Verschragen (Project manager) – SWOV Institute for Road Safety Research	Final revisions and checks
21/12/2020	Final deliverable	Prof. Dr. Nicole van Nes (Project coordinator) – SWOV Institute for Road Safety Research	Approved
29/06/2021	Final updated draft	Ingrid van Schagen (QA officer), Ernst Verschragen (Project manager) – SWOV Institute for Road Safety Research	Update after P1 review
30/06/2021	Final updated deliverable	Michiel Christoph (Technical project coordinator) – SWOV Institute for Road Safety Research	Approved

Table of contents

List of Abbreviations

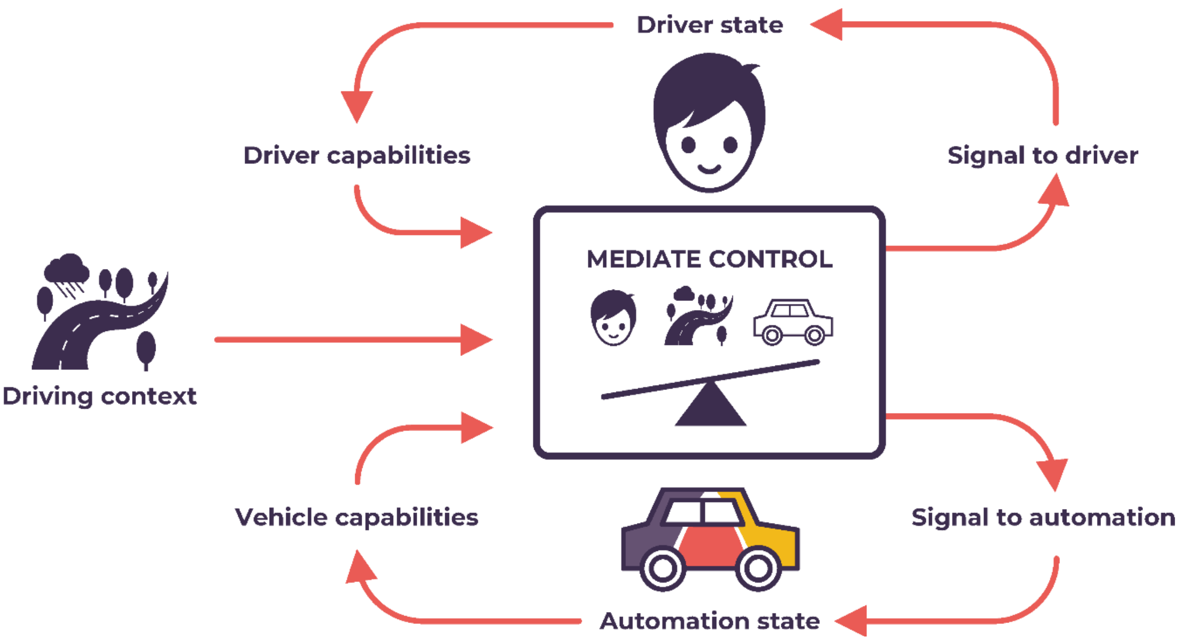
CM	Continuous Mediation (automation level)
HMI	Human Machine Interface
KPI	Key Performance Indicator
KSS	Karolinska Sleepiness Scale
NDRT	Non-Driving Related Task
SB	Driver Standby (automation level)
TTAF	Time To Automation Fitness
TTAU	Time To Automation Unfitness
TTDC	Time To Driver Comfort
TTDD	Time To Driver Discomfort
TTDF	Time To Driver Fitness
TTDU	Time To Driver Unfitness
TtS	Time To Sleep (automation level)
UC	Use Case

About MEDIATOR

MEDIATOR is a 4-year project led by SWOV. It started in May 2019. MEDIATOR will develop a Mediating system for drivers in semi-automated and highly automated vehicles, resulting in safe, real-time switching between the human driver and automated system based on who is most fit to drive. MEDIATOR pursues a paradigm shift away from a view that prioritises either the driver or the automation, instead integrating the best of both.

Vision

Automated transport technology is developing rapidly for all transport modes, with huge safety potential. The transition to full automation, however, brings new risks, such as mode confusion, overreliance, reduced situational awareness and misuse. The driving task changes to a more supervisory role, reducing the task load and potentially leading to degraded human performance. Similarly, the automated system may not (yet) function in all situations. The objective of the Mediator system is to intelligently assess the strengths and weaknesses of both the driver and the automation and mediate between them, while also taking into account the driving context.



The Mediator system will constantly weigh driving context, driver state and vehicle automation status, while personalising its technology to the drivers' general competence, characteristics, and preferences.

MEDIATOR will optimise the safety potential of vehicle automation during the transition to full (level 5) automation. It will reduce risks, such as those caused by driver fatigue or inattention, or on the

automation side imperfect automated driving technology. MEDIATOR will facilitate market exploitation by actively involving the automotive industry during the development process.
To accomplish the development of this support system MEDIATOR will integrate and enhance existing knowledge of human factors and HMI, taking advantage of the of expertise in other transport modes (aviation, rail and maritime). It will develop and adapt available technologies for real-time data collection, storage and analysis and incorporate the latest artificial intelligence techniques, such as deep learning.

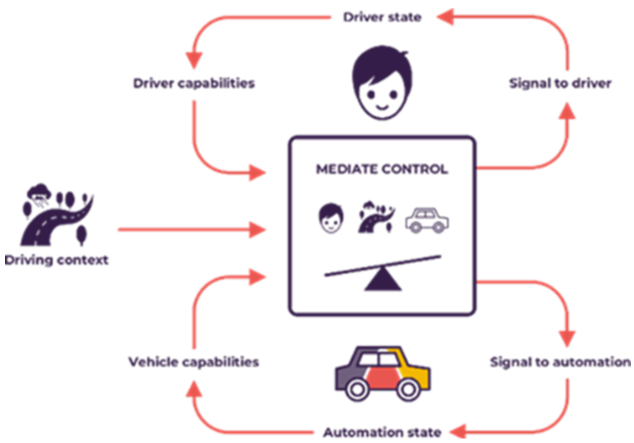
Partners

MEDIATOR is being carried out by a consortium of highly qualified research and industry experts, representing a balanced mix of top universities and research organisations as well as several OEMs and suppliers. The consortium, supported by an international Industrial Advisory Board and a Scientific Advisory Board, also represents all transport modes, maximising input from, and transferring results to, aviation, maritime and rail (with mode-specific adaptations).

Executive summary

Vehicle automation is developing rapidly, with substantial potential to improve mobility effectivity and efficiency of road transport. Vehicle automation can also have a large effect on safety, since it reduces the influence of human fallibility. However, at the same time, vehicle automation is likely to introduce new risks, particularly during the transition phase to full automation when vehicle control is shared by human drivers and automation. By sharing control, the task of the driver changes from an active to a more passive, supervisory role. Unfortunately, humans are not very good in monitoring and supervising tasks. It reduces the workload of the primary driving task and if reduced too much, this might lead to boredom, reduced attention and situational awareness, mode confusion, loss of skills and, eventually, to degraded performance with responses that are too late or even non-existent. In these cases, it will be dangerous to transfer the control from the automation system to the human driver. However, in other situations humans are more reliable than the automation system. In unusual, unexpected, or complex situations, humans are often better than systems. A human driver could quickly find a more tailored, unique solution while an automation system must rely on (still) inappropriate rules or algorithms. In those case, it would be safer to have the human driver in control rather than the system.

MEDIATOR aims to develop an intelligent ‘mediator’ support system, enabling safe, real-time switching between the human driver and automation, integrating the best of both. The figure below illustrates the basic principles of the Mediator system. It shows that the system continuously and in real time monitors and weighs the information about the driving context, the driver state, as well as the automation status, while taking account of the general competences of the driver and the vehicle capabilities.



Schematic summary of the functioning of the Mediator system

By weighing these elements, the Mediator system will determine how and to what extent the human or the machine should be in control, enabling safe, comfortable, real-time switching between the two by a reliable, understandable and user-friendly human machine interface (HMI). To allow for timely announcement of a transfer of control the system will not only assess the current situation but also make a prediction of the status in the near future. For reasons of acceptability the system will take account of driver’s preferences, however without compromising safety.

This document, which is the final deliverable of task 1.4, describes the functions of the Mediator system in more detail and derives a set of functional requirements from them. The information presented in this document is based on the work done on state of the art and preliminary requirements in task 1.1 (Christoph et al., 2019), driver state in task 1.2 (Borowsky et al., 2020), automation state in task 1.3 and Human Machine Interaction in task 1.5 (van Grondelle et al., in preparation).

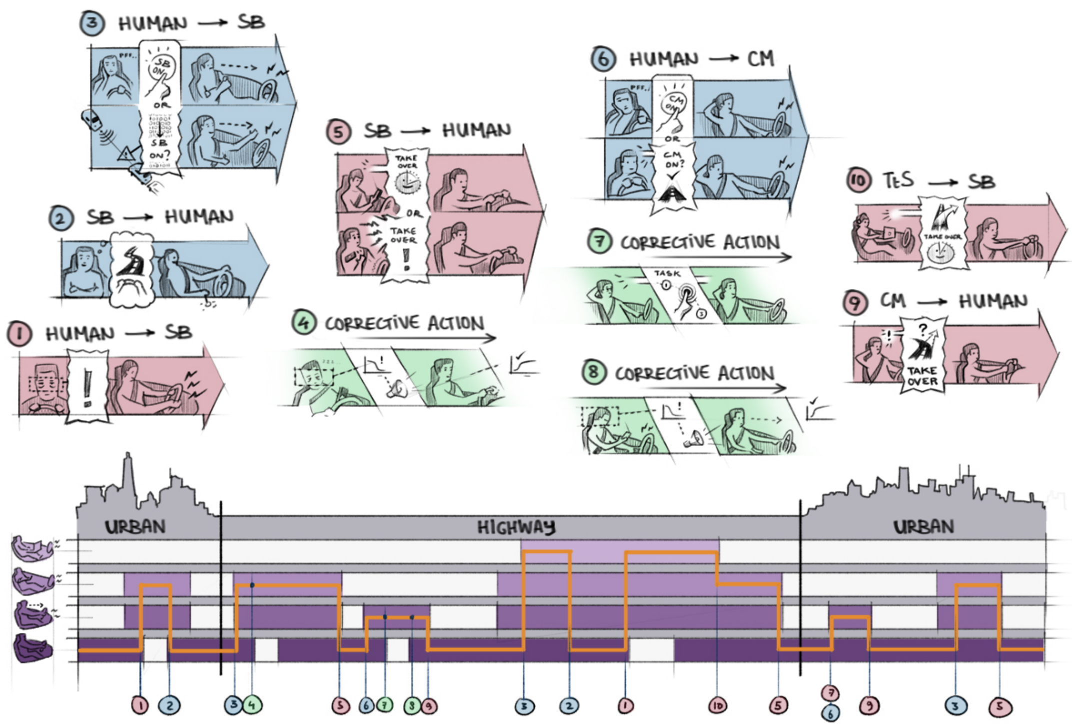
Three levels of automation

The Mediator system will be developed for three automation levels, that reflect the required activity of the driver and the type of support needed:

1. **Continuous Mediation (CM) – Driver In the Loop** describes 'assisted driving'. Drivers are responsible but supported by the automation. The automation generally performs the active control tasks, while the driver has a monitoring task. Challenges in this level of automation are *creating mode awareness* and supporting the driver with their part of the driving task by *creating an optimal task load*.
2. **Driver Standby (SB) – Short Out of the Loop** describes 'conditional automation'. Drivers can be out of the loop for a short time but must remain 'on standby' to take back control when needed. Challenges here are related to *regaining driver fitness* and balancing the time until the automation or driver becomes unfit, making sure always one is fit enough for the driving task. This challenge extends to the HMI challenges *of communicating these time budgets* and mediating *comfortable and safe takeovers* over a relatively short time span.
3. **Time to Sleep (TtS) – Long Out of the Loop** describes 'highly level automation'. Drivers can be out of the loop for long periods of time and truly immerse themselves in non-driving related tasks and even fall asleep. Challenges in this level of automation are to *bring the driver back in the loop after full disengagement* and to predict when this will be required long enough in advance.

Ten use cases

During the project, for these three levels of automation, the Mediator system will be developed to cover ten specific scenarios or use cases. These use cases include both urban and motorway scenarios and focus on the detection and mitigation of driver degraded performance due to both distraction and fatigue. The ten use cases are visualised below, with on the bottom a typical trip from an urban situation via a motorway to another urban situations, with several switches between automation levels and corrective or preventive actions to improve or maintain driver fitness while driving within one level of automation.



Visualisation of the ten use cases

The driving context, e.g., weather and traffic conditions and the presence of other road user types, will vary between and within use cases, depending on the exact aim and setup of related experiments.

Focus points

The human-machine interface (HMI) is of crucial importance for a Mediator system, both from a safety and an acceptability point of view. The HMI must not only be usable throughout the various situations, but at the same time it must address issues of trust, transparency, and personal preferences. Four focus points of the Mediator HMI were identified:

Different types of takeover take place from driver to automation and vice versa and can be subject to different levels of necessity (e.g., comfort vs safety related takeovers) and different levels of urgency. The HMI should be able to handle all these different takeover situations.

In the conditional automation and the high-level automation conditions drivers must be informed, clearly and at all times, how long the current level of automation will last and when they are supposed to take over. The continuous, transparent communication about the **time budget** to the driver by the HMI is a key ‘preventive’ activity of the Mediator system.

In the **continuous mediation** automation level, the HMI must help to combat underload-induced driver distraction and fatigue which is likely to occur in situations where people mainly have a monitoring task. A focus of the current project is to research and develop ideas of balancing the driver workload for optimal task performance and driver comfort for this level of automation.

Personalisation will be an important aspect of the Mediator system and the related HMI. Personalisation can refer to the detection of driver states, the optimisation of decisions, and the communication. Personalisation helps to 'automatically' account for possible differences between drivers due to, for example, age, gender, or cultural differences.

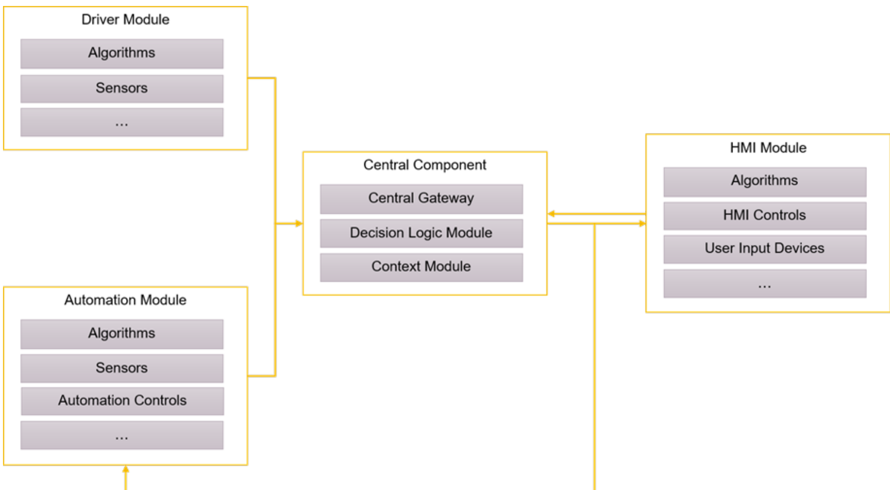
Resulting high level functional requirements of the Mediator system

At a high level the functional requirement of the Mediator system can be structured chronologically as 'event detection', 'action selection' and 'action execution': first an event that requires a Mediator action is detected, then the appropriate action is selected based on safety and comfort requirements, and finally, the selected action is executed. In more detail the high-level system functions were summarised as follows.

- Detect: user requests, current automation level
- Estimate: time to automation/driver (un)fitness, time to driver (dis)comfort
  - Estimate and predict driver fatigue and distraction
  - Predict relevant context changes
- Decide: who is fittest to driver, considering driver safety and driver comfort
- Select: action for optimal driving conditions related to safety and comfort
  - For takeovers and corrective actions, determine available time scale and necessity of action (urgency level)
- Execute: selected action
  - Promote optimal cooperation between driver and automation through direct and indirect communication with the driver.
- Monitor: action execution
- Adjust: action or urgency level during execution if needed

The four modules of the Mediator system

The Mediator system consists of the integration of four different modules or components: the driver module, the automation module, the HMI module, and a central component, as shown in this figure.



Schematic overview of the four modules of the Mediator system

Driver module: has multiple functions. It 1) estimates the time to driver (un)fitness and time to driver (dis)comfort used to decide who is fittest to drive, 2) provides information on possible actions for improving driver fitness or comfort 3) estimates the corresponding expected fitness improvement and available time scale, and, finally, 4) describes the reason for the reduced fitness (e.g., fatigue or distraction). Instruments needed include in-vehicle cameras an intelligent steering wheel and/or seatbelt equipment which is currently under development.

Automation module: ensures the interface between the Mediator system and the available or operational vehicle automation functions. It provides information on current automation state and can adjust automation state if needed. It also provides data about the driving context, e.g., weather related of traffic related.

HMI module: communicates between the vehicle related systems, including the Mediator system, and the human driver. Among other things it facilitates negotiations between driver and automation, guides control transfers between driver and automation (takeovers), informs which systems are switched on or switched off, executes actions to increase or maintain driver fitness, and takes account of driver preferences. The HMI should have high usability and transparency for its users and help to improve driver comfort and safety.

Central component: contains the decision logic and context modules and serves as a central gateway to transfer information between all modules in the Mediator system:

- The **decision logic module** translates information about the state of the driver and the automation from the respective modules into a Mediator action requests. In this process the module detects events that require Mediator actions and selects appropriate actions based on information about past, current and expected situations. These requests are then sent to either the HMI or the automation module that is responsible for the execution of the actions and monitored such that adjustments can be made if needed.
- The **context module** receives context information from different sources, including the automation module and in-vehicle equipment for status monitoring. The module integrates this information into specific context characteristics requested by other modules, mainly the driver module, the HMI module, and the decision logic.

Ethics and validation

As the Mediator system can affect society, also ethical considerations should be taken into account in its design. Important ethical considerations lead, for example, to requirements on system transparency, accountability and validity and the autonomy of humans.
To evaluate how effectively the Mediator system is achieving the objectives proposed throughout this document, a set of key performance indicators is defined. This initial list of indicators can be used in the validation of the system.

Conclusion

The work presented in this document leads to an overview of a set of functional requirements for the Mediator system. These functional requirements are used to guide design and development of the Mediator system, subject to the constraints given by development time, budget, and available prototype platforms.

1. Introduction

This deliverable provides an overview of the Mediator system, its scope and the corresponding functional requirements as identified within Work Package 1. The main focus of Task 1.4, of which this deliverable is the final product, was to integrate the research done in Tasks 1.2, describing the driver module, 1.3, describing the automation state module and 1.5, describing the human machine interface (HMI). A second focus was to take the first steps towards the design of the decision logic module, which connects all modules into one Mediator system. The final result of this integration is the overview of the functional requirements for the Mediator system that feed into the next stages of the project.

Vehicle automation has the potential to improve driving safety and driver comfort. However, as with any successful human-machine system, the interaction between the two should be designed to support this. To this aim, the Mediator system mediates between the driver and the automation on who is fittest to drive and supports the driver during his or her driving task. Both comfort and safety are achieved by taking into account both the driver capabilities and needs as well as the automation capabilities at each moment in time and predicting them for the near future.
The Mediator system focusses on vehicles that have up to SAE level 4 automation functions, i.e. on vehicles where a driver is still required for at least parts of the trip. In these situations, known driving challenges related to human factors, such as driver fatigue and distraction, are still relevant and can be partially solved by vehicle automation. In other cases, an intelligent HMI design could support the driver in his or her driving task and reduce driver fatigue and distraction. With new automation functions, however, also new challenges arise, such as mode confusion, overreliance and automation surprise (Saffarian, 2012). The Mediator system is therefore designed to also cope with these aspects of driving with vehicle automation. To this end, the Mediator system focusses on preventing safety critical events and driver discomfort by predicting future changes in the driving related conditions and preparing and supporting the driver to manage them.

First steps to further define the Mediator functions and identify knowledge gaps were taken in Task 1.1 of the MEDIATOR project. This was followed by research work to fill these knowledge gaps in Tasks 1.2, 1.3 and 1.5.

The main purpose of this document is to present the set of Mediator system *functional requirements*, based the work done in Work Package 1. These requirements serve as a basis for the next stages of the project, the development and evaluation of the Mediator system. The requirements specify the system to be built and developed (WP2) and the criteria for evaluation (WP3). The document provides an overview of the Mediator system, its scope, ethics and validation considerations and concludes with the set of identified functional requirements. At the end of the project, an update version of the requirements will be prepared to include the new insights gained during the project.

In Chapter 2 first the narrowed down scope of the MEDIATOR project is presented in the shape of 10 use cases and main focus points. Second, the resulting high-level Mediator functions are briefly described.

To provide an overview of the complete Mediator system, Chapter 3 presents a summary of the research performed in Tasks 1.2, 1.3 and 1.4, in combination with the proposed integration steps based on several meetings with the partners involved. Section 3.1 describes the different functions of the driver module to assess and improve both driver safety and comfort and the corresponding algorithms and look up tables that will be used to implement them. In Section 3.2 a brief overview of the automation module functions is presented. Section 3.3 summarises possible outputs of the context module, which can be used as input to functions in other modules. The main functions of the HMI module, which is responsible for communication between the driver and the Mediator system, are described in Section 3.4. Finally, Section 3.5 elaborates on the main functions of the decision logic. This module is responsible for detecting situations in which Mediator actions are required, and selecting, monitoring and possibly adjusting the corresponding appropriate actions.

As the Mediator system can affect human safety and well-being, also ethical principles should be considered in its design. Chapter 4 therefore outlines existing ethics recommendations from the fields of artificial intelligence and vehicle automation and applies them to the Mediator system.

While the validation of the Mediator system will be performed in later stages of the project, insight into how the Mediator system and subsystems could be evaluated, including the key performance indicators that can be used, can already guide some of the design choices, which is explained in Chapter 5.

From information in both chapters 4 and 5 also several functional and non-functional requirements are derived. So, finally, in Chapter 6 the set of functional requirements for the complete Mediator system as identified in this stage of the project is presented. These functional requirements provide input to WP2, where they are used to guide design and development of the technical Mediator components and overall system, subject to the constraints given by development time, budget, and available prototype platforms.

2. Scope of MEDIATOR Project

While the Mediator system, once on the market, should cover all driving scenarios, this is scope is too broad for the initial phases of its development, i.e., this MEDIATOR project. Instead, a set of use cases have been defined with the aim of covering the most relevant and characteristic scenarios for the Mediator system, while also showing the broad range of scenarios in which the Mediator system can aid drivers of automated vehicles. In this chapter, the set of use cases, as well as some special focus points of the Mediator system and the corresponding high-level functions, are described.

In D1.1 (Christoph et al., 2019) high-level use cases were defined that distinguish between different levels of automation. In the current document they will from now on be referred to as automation levels, to distinguish them from the use cases described here. The automation levels differ somewhat from the commonly used SAE levels of automation. The SAE levels are described from an engineering perspective, while the levels chosen in the project aim to approach levels of automation from a human-centred perspective. To this end, the automation levels focus on needs and challenges of the driver and the support needed. The three levels of automation are described as follows:

4. **Continuous Mediation (CM) – Driver In the Loop** describes assisted driving, where the driver is responsible but is supported by the automation. There is a division of tasks between driver and automation: the automation generally performs all active control tasks, while the human driver takes on the task of monitoring and maintaining sufficient situation awareness. Challenges in this level of automation are *creating mode awareness* and supporting the driver with their part of the driving task by *creating an optimal task load*.
5. **Driver Standby (SB) – Short Out of the Loop** describes conditional automation, where the driver can be out of the loop for some periods of time but can be requested to takeover anytime when the automation becomes unfit. The driver has to remain 'on standby' to take back control. Challenges in this automation level are related to *regaining driver fitness* and balancing the time until the automation or driver becomes unfit, making sure always one is fit enough for the driving task. This challenge extends to the HMI challenges *of communicating these time budgets* and mediating *comfortable and safe takeovers* over a relatively short time span.
6. **Time to Sleep (TtS) – Long Out of the Loop** describes driving with high automation, where the driver can be out of the loop for long periods of time. Drivers can then truly immerse themselves in non-driving related tasks and even fall asleep. Challenges in this level of automation are to *bring the driver back in the loop after full disengagement* and to predict when this will be required long enough in advance.

In the next sections first the main focus points of the MEDIATOR project are discussed. Next, the 10 use cases, which together describe the scope of the project, are presented. From these use cases initial high-level functions of the Mediator system are derived.

2.1. Focus points

The general focus of the MEDIATOR project is to increase the safety at higher levels of automation. Generally, the focus is on how the system should react to unsafe situations and emergency scenarios. In this project, we take a different perspective. The main focus is on *preventing* unsafe situations, anticipating on the situation and avoiding unsafe situations to evolve. Also offering driver *comfort* is an important focus point, as the system can only be effective if it is appreciated by the driver. Hence, the system also suggests actions to increase the comfort of the driver. This focus is reflected in the use cases selected: they do not reflect extreme situations or emergency stops, but rather common everyday driving situations where prevention of unsafe and uncomfortable conditions is the main goal. The selected use cases are described in Section 2.2.

An important challenge for the MEDIATOR project is to develop an **integrated HMI** (Human Machine Interface) that is usable throughout the various situations and at the same time addresses the associated issues of trust, transparency, and personal preferences well. Four main focus points that eventually should be handled by such an HMI were identified: Takeovers, time budget communication in full automation, continuous mediation in partial automation, and personalisation.

Takeovers can occur from driver to automation and vice versa and can be subject to different levels of necessity (e.g., comfort vs safety related takeovers) and different levels of urgency. The HMI should include a procedure that can be adapted to all these different takeover situations.

Time budget refers to the principle to show to the driver, clearly and at all times, how long the current level of automation will last. This is an important concept for all situations and needs to be simultaneously and coherently addressed by all sub modules of the Mediator system. The available time that the current driver (human or automation) is still fit and the other becomes unfit will need to be estimated. These estimations and their confidence intervals need to be dealt with when deciding upon a Mediator action. In terms of human factors and HMI, these time budgets should be made (explicitly or implicitly) very transparent to the human driver. The continuous, transparent communication of the time budget to the human via the HMI is viewed as a key "preventive" activity of the Mediator system.

Continuous mediation refers to the challenge that arises during driving in this level of automation, where underload-induced driver distraction and fatigue are known to occur. A focus of the MEDIATOR project is to research and develop ideas of balancing the driver workload for optimal task performance and driver comfort while driving in CM.

Personalisation will be an important aspect of the Mediator system. Personalisation can be applied to the detection of driver states, the optimisation of decisions made by the Mediator system and the HMI. By having personalised, individual algorithms for such functions, the project 'automatically' accounts for possible differences between different people due to, for example, age, gender, or cultural differences.

2.2. Use cases

A total of ten use cases were developed to define the scope of the MEDIATOR project. These use cases describe specific scenarios in which the Mediator system is expected to act to improve the safety and comfort of the driver. Some general scoping decisions were made that apply to all these scenarios:

- Only urban and highway scenarios are considered, i.e., exclude driving on rural roads.
- Only detection/mitigation of specific types of driver distraction and fatigue:
    - Distraction: focus on eyes off road (and hands-off steering wheel)
    - Fatigue: focus on detection of both 'recoverable' fatigue (which is typically task-related, e.g., due to underload or overload) and 'non-recoverable' fatigue (which is typically sleep-related, e.g., due to sleep deprivation). But the use cases will not focus

on mitigation actions/strategies for such strong degradation due to severe sleep-related fatigue (e.g., waking up the driver after sleeping). During the complete trip, the human driver is awake and seated in the driver seat.

The ten use cases are visualised in Figure 1. As both safety and comfort are important considerations for the MEDIATOR project, these use cases are labelled as mainly safety (red or green) or mainly comfort (blue) related. A distinction is also made between safety related use cases that describe takeovers between automation levels (red) and those that describe driving within one level of automation (green). In the latter type there is the important distinction between "preventive" actions that the Mediator system will take and "corrective" actions. Both are related to human degraded performance. Corrective actions attempt to correct degrading human performance, such as issuing warnings with different intensity levels. Preventive actions instead aim to prevent human fitness from degrading to a state in which correction becomes necessary.

A timeline of a trip that includes all use cases is shown in the bottom of Figure 1. The top row indicates that the drive starts in an urban environment, then continues on a highway, and finally finishes again in an urban environment. The levels of automation that are available at each point in time are shown in the purple bars, while the orange line indicates the actual automation level that is active. The bottom bar shows manual driving, which is marked purple when the human driver is fit to drive. The three bars above show the automation levels CM, SB and TtS, respectively and are marked purple if the corresponding automation fitness for that level is sufficient.

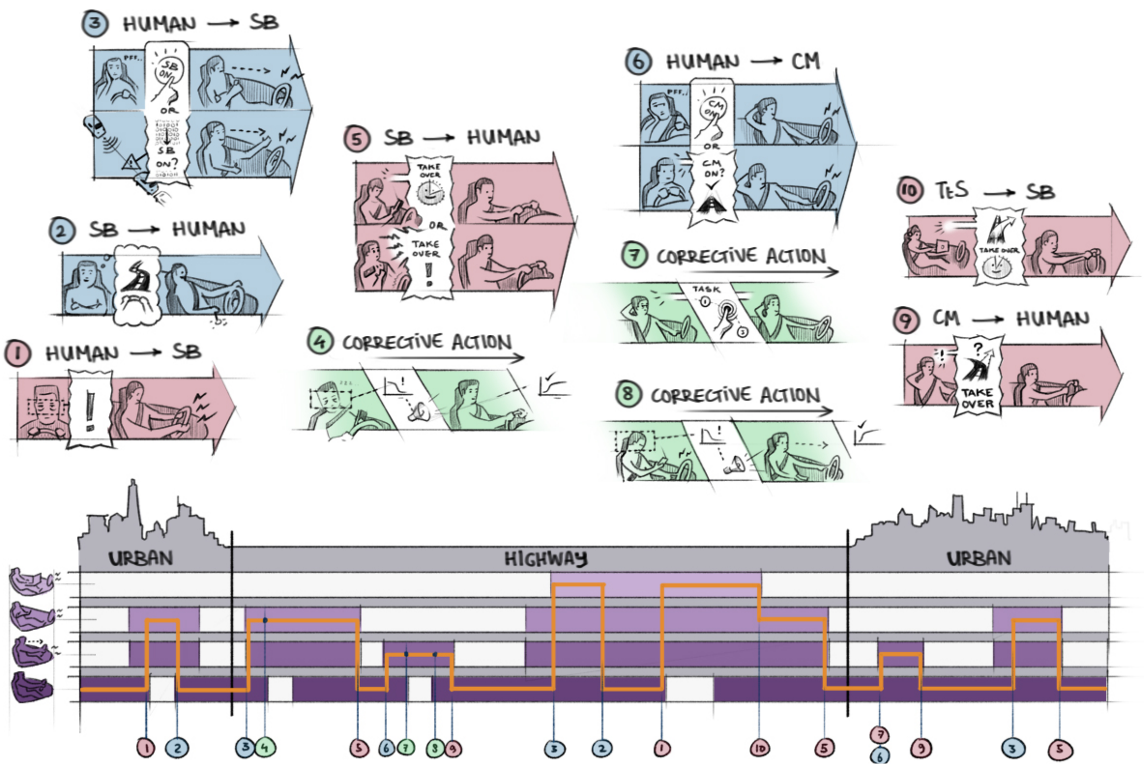


Figure 1: Mediator Use Cases

A more detailed description of each of the ten use cases is given below:

1. Mediator system initiates takeover (human to automation): Degraded human fitness, caused by either drowsiness (a) or distraction (b), is detected by the Mediator system. The system reacts by initiating a takeover to automation.
2. Driver takes back control: The driver uses the HMI to indicate a desire to take back. The Mediator system reacts by confirming that the driver is fit enough to drive and guiding the takeover.
3. Comfort takeover (human to automation): Either the driver (a) or the Mediator system (b) initiates a takeover from human to automation.
   a) The driver indicates via the HMI that he/she is not motivated to drive. The Mediator system reacts by confirming the automation fitness and guiding the takeover.
   b) The Mediator system detects an event, such as receiving a text message or an upcoming traffic jam, from which it concludes that the driver comfort could be improved. The system reacts by suggesting a takeover to automation.
4. Corrective Action (SB): While driving in SB the human driver becomes drowsy. The Mediator system reacts by initiating an action to improve the driver fitness and monitors the effect.
5. Mediator initiated takeover (automation to human): A planned (a) or an unplanned (b) takeover from automation to human is initiated by the Mediator system.
   a) The automation indicates that the current route leads to automation unfitness as it will leave its operational design domain. The Mediator system reacts by preparing the driver for and guiding the driver through a non-urgent takeover.
   b) The automation indicates that its fitness is rapidly degrading and can soon no longer perform the driving task. The Mediator system reacts by informing the human driver and guiding the urgent takeover.
6. Comfort CM switch on: Either the driver (a) or the Mediator system (b) switches on driving in CM.
   a) The driver indicates via the HMI that he/she is not motivated to drive. The Mediator system reacts by confirming the automation fitness and switches on CM.
   b) The Mediator system detects sufficient fitness for driving in CM from which it concludes that the driver comfort could be improved and reacts by suggesting switching on CM.
7. Preventive Action (CM): While driving in CM, the driver is supported by the Mediator system in performing the monitoring task. The system does this by trying to prevent underload and keeping the driver in the loop.
8. Corrective Action (CM): While driving in CM, degraded driver fitness is detected by the Mediator system. The system reacts by initiating a corrective action to improve driver fitness.
9. CM shuts off instantly: While driving in CM, the automation fitness degrades and automation can no longer perform its driving task. The Mediator system reacts by communicating to the driver that CM is switching off.
10. Smooth transition from TtS to SB: while driving in TtS the driver is fully disengaged from the driving task when the automation indicates that the current route will leave the operational design domain. The Mediator system detects sufficient automation fitness for driving in SB and reacts by informing the driver that SB will be switched on and subsequently monitors the required driver fitness.

Driving Context

The driving context describes under which environmental conditions each of the ten use cases occur. This can, for example, include weather and traffic conditions and the presence of certain types of other road users. The exact driving context of the ten use cases is not fixed. Instead, the specific driving context variations to be considered *depend on the specific focus* of the experiments to be done and the specific sub-use case(s) they are related to. However, for feasibility reasons, during the MEDIATOR project we will *limit* the number of different driving context variations to be considered for such experiments and related prototype systems to only a few, namely those that are most relevant for those experiments and specific related sub-use cases.

As mentioned in the previous section, the focus is on highway and urban environments, and rural roads are thus excluded. In both situations high and low workload scenarios will be considered. In urban scenarios, situations with and without vulnerable road users, such as pedestrians and cyclists, will be included. Specific driving context variations to be considered may include different traffic situations (little traffic, much but free-flowing traffic, congested traffic, mixed road user types vs. similar road user types) and different environmental situations (day vs. night, good visibility vs. bad visibility, rain vs. no rain).

In this stage of the project no fixed driving context variations have been decided yet. This decision will be made in later stages, when the details and requirements of specific experiments and prototype (sub)systems have been worked out. The various driving context variations will be aligned as much as possible (while allowing for the necessary differences, as described above). Such alignment will facilitate comparisons between the results of experiments and evaluations and facilitate the development of simulators and prototype systems.

2.3. Mediator system functions

The use cases provide insight into what is expected of the Mediator system in different scenarios. In general, these functions can be structured chronologically as event detection, action selection and action execution. Where first an event that requires a Mediator action is detected, then the appropriate action is selected based on safety and comfort requirements and finally, the selected action is executed. In the next sections these three main functions are further explained.

2.3.1. Event detection

Each of the ten use cases (UC) described in Section 2.2 starts with a stable driving situation in either manual driving or the automation levels Continuous Mediation (CM), Driver Standby (SB) or Time to Sleep (TtS). During all uses cases an event occurs after some time. Such events entail that either the *driver* (UCs 1, 4 and 8) or the *automation* (UCs 5, 9 and 10) shows *degraded performance* and is predicted to become unfit in the near future, that the *driver comfort* is predicted to reduce (UCs 3b and 6b) or that a *driver request* for takeover is detected (UCs 2, 3a and 6a). In UC 7, just before the use case starts, the automation level changes to CM. This event, the start of driving in CM, should also be detected as it needs to trigger a preventive action. A first Mediator system function is thus to detect the following events: driver degraded performance, automation degraded performance, driver expected discomfort, driver takeover requests and new automation level.

In order to act on these events in a safe, comfortable and thus timely manner, fitness and comfort rather than being *detected*, should be *predicted* for the near future. Within the scope of the MEDIATOR project, only driver fitness related to (visual/manual) distraction and fatigue will be estimated and predicted. Such fitness prediction can also vary between contexts. For example, a driver might be getting slightly drowsy but still able to drive on a quiet road without junctions, while in this same fitness state the driver might be unfit to drive in a hectic city environment. To predict driver and automation fitness and discomfort, also relevant context information should be predicted.

2.3.2. Action selection

Following the described events, one of several interventions is initiated by the Mediator system. Either a *takeover procedure* from automation to human or vice versa, or a *fitness improvement procedure* is started. Based on who is or will be fittest to drive, the Mediator system will decide on which action to take. Both safety and driver comfort should be taken into account by the Mediator system when taking such decisions. To make these decisions, both driver and automation fitness measures need to be comparable. To this end, in D1.1, Christoph et al. (2019) defined the time to driver/automation (un)fitness variables. These variables are comparable and include not only the driver fitness state but also its prediction.

The takeover process can differ depending on several factors. Comfort takeovers, for example, are not mandatory but are suggested to improve driver comfort. Takeovers due to driver unfitness, though, should be strongly suggested. Nevertheless, they can be overwritten if the driver fitness shows improvement. Takeovers due to automation unfitness should generally not be allowed to be overwritten by the driver, as it is unlikely that the automation fitness suddenly improves. These takeovers due to automation fitness can also differ depending on whether they were planned, where generally the takeover time is longer, or unplanned, where a short takeover time applies. Any takeover procedure should thus take into account the available takeover time and necessity of the takeover. Additionally, as the takeover process takes time, its progress should be monitored and adjusted or cancelled when the situation changes, such as when during a takeover from driver to automation the driver fitness is improving.

One type of fitness improvement procedures are the corrective actions, which aim to improve driver fitness when degraded fitness has been detected. These can occur as a start of a takeover from automation to human, but also while driving with automation or even during manual driving. The MEDIATOR project scope includes specific instances of such actions during the automation levels CM and SB. In these cases, the required fitness differs, i.e., during CM the driver needs to be fit enough for the monitoring task, while during SB the driver needs to be fit enough to take over control within a limited time span when requested. The corrective actions and/or the decision to initiate them should take such differences into account. For example, in CM, the driver might be corrected when any distraction is detected, while in SB the driver will only be corrected if this distraction leads to an estimated takeover time that exceeds the minimal time to automation unfitness. In this case also the procedure might differ. In CM the action might be called a success when the driver has his eyes back on the road, while in SB the action might be assumed successful when the driver changed his non-related driving task (NDRT). Both the takeover procedure and the corrective action require time; hence its progress should be monitored. If the progress is not as expected, the Mediator system should adjust its strategy.

Another fitness improvement procedure, the preventive actions, occur in UCs 7 and to some extend in 10, where the driver is not actively involved in the driving task, while (s)he is expected to either monitor (UC7) or supervise (UC10) the automation. Such tasks can be demanding and sometimes unclear to the driver. To aid the driver with these tasks and prevent degraded driver fitness when driving with automation, the Mediator system should initiate appropriate preventive actions. These actions can, for example, include creating an appropriate level of workload and involvement, improving mode awareness, and creating an activating in-vehicle ambience.

2.3.3. Action execution

To achieve the desired high levels of safety and driver comfort, the communication with the driver should be such that an optimal cooperation between human and automation is achieved. For example, takeover procedures and corrective actions should be communicated in an understandable fashion using HMI controls that express the appropriate level of urgency. In its communication to the driver, the Mediator system should increase mode awareness and a level of understanding of the system that reduces automation surprises and overreliance and increases driver comfort and trust. And, as drivers' preferences differ, personalisation should be implemented when feasible.

2.3.4. Summary of high-level Mediator function requirements

The high-level system functions for the use cases in the MEDIATOR project can be summarised as follows:

- Detect: user requests, current automation level
- Estimate: time to automation/driver (un)fitness, time to driver (dis)comfort
    - Estimate and predict driver fatigue and distraction
    - Predict relevant context changes
- Decide who is fittest to driver, taking into account driver safety and comfort
- Select action for optimal driving conditions related to safety and comfort
    - Takeovers and corrective actions: Determine available duration and necessity of action (urgency level)
- Execute selected action
    - Promote optimal cooperation between human and automation through direct and indirect communication with the driver.
- Monitor action execution
- Adjust action or urgency level during execution

In the next chapter the different system modules responsible for (parts of) these functions are described.

3. The Mediator System

While the previous chapter provided some high-level Mediator system functions, in this chapter a more detailed look into the Mediator functions within different modules is provided. The Mediator system is generally subdivided into five main modules: the driver module, the automation module, the context module, the human machine interface (HMI) module and the decision logic module. In this chapter the main functions of each of these modules are further explained.

In Figure 2 an overview of the different modules of the Mediator system is presented. The central component contains both the decision logic module, where driver and automation fitness are compared and Mediator actions are selected, as well as the context module, where all context relevant information is gathered and transformed into required context variables. The central component also contains a central gateway, which serves as a throughput channel for any additional variables that need to be exchanged between modules. The driver and the automation module provide input to this central component in the form of fitness, intervention and state related variables. These modules contain both software and hardware related components. The communication to the driver is done via the HMI module, where again software and hardware related components are present. This module also transfers some information to the central component, such as driver inputs. The communication with the automation is instead done via the automation module. In the next sections the functions within each module are further explained.

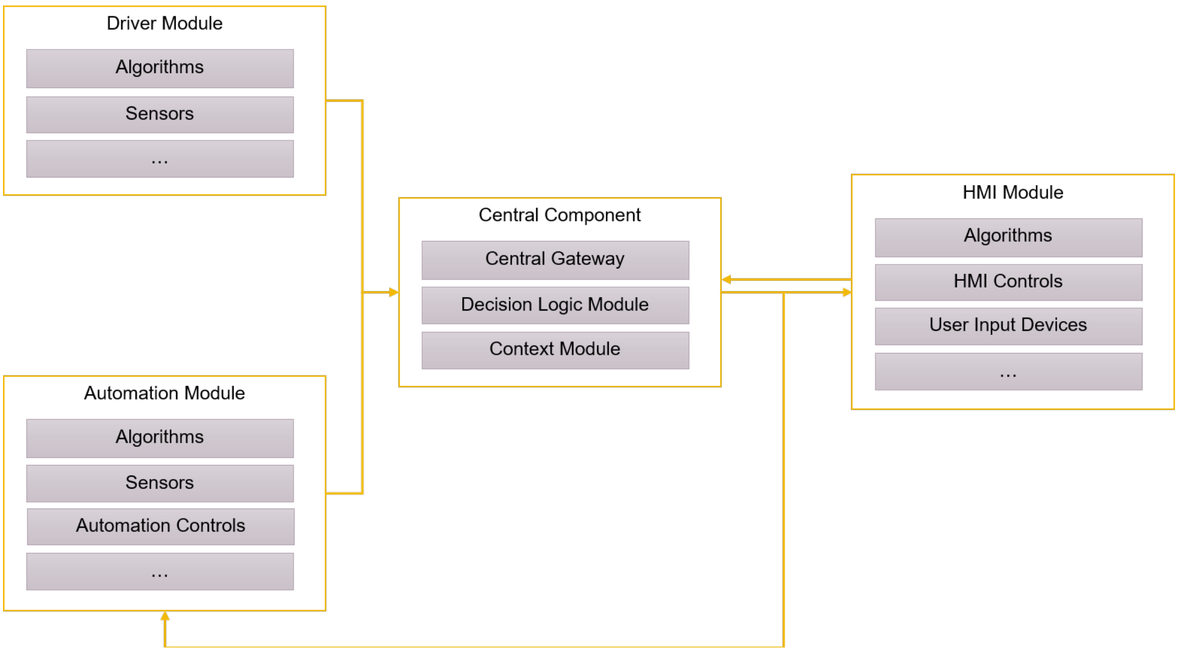


Figure 2: Mediator system overview

3.1. Driver Module

The main function of the driver module is estimation of the time to driver (un)fitness and time to driver (dis)comfort which are used to determine who is fittest to drive. Additionally, to select the appropriate actions, this module provides information on possible interventions that could improve driver fitness or comfort together with an estimate of the corresponding expected fitness improvement and available time scale. Finally, to support the decision making, information on the cause of reduced fitness, the driver state class, which describes the reason for the reduced fitness, is also outputted. More detailed information on the research work and algorithms mentioned in this chapter can be found in the Mediator deliverable of task 1.2 (Borowsky,2020).

3.1.1. Time to driver (un)fitness

The *time to driver fitness* is defined as the estimated time before a driver is able to safely perform the manual driving task and is most relevant in SB or TtS where the driver is not expected to be involved in the driving task at all times. The *time to driver unfitness* is defined as the estimated time until the driver is no longer able to safely perform the manual driving task and is most relevant while driving manually or in CM, where the driver is expected to continuously be involved in the driving task. Within the MEDIATOR project scope, driver fitness is estimated from levels of driver fatigue and distraction, where distraction detection is limited to visual and manual distraction. The algorithms used to calculate driver fitness variables will first estimate driver fitness levels in terms or fatigue and distraction, then predict these levels in the near future and finally combine them into one overall measure of time to driver fitness or unfitness.

3.1.1.1. Fatigue

Two preliminary *fatigue detection* algorithms were developed using real world driving data from sleepy and awake drivers. Both algorithms estimate the *Karolinska Sleepiness Scale (KSS) score* that was obtained during the experiment, ranging from 1 (extremely alert) to 9 (very sleepy). The first algorithm (Algorithm A in Figure 3 in Section 3.1.6) uses the physiological measures from ECG, heart rate and respiratory sensors to estimate the KSS score. The second algorithm (Algorithm B in Figure 3) instead extracts head, face and eye patterns from face video to estimate the KSS score using individualised models. Both algorithms showed promising results and, as a next step, they will be merged to improve accuracy, reliability and feasibility of the overall Mediator fatigue detection algorithm. During manual driving, the overall algorithm can also make use of measures of driving performance to improve accuracy. For example, information on *sleep and driving history* as well as *time of day* could be added to improve the accuracy of the algorithm.

The Mediator fatigue detection algorithm will require different sensor inputs to estimate the driver's KSS score. During manual driving the *heart rate* can be measured via an intelligent steering wheel[1]. In other situations, the heart rate can be measured via a wearable device such as a fitness tracker and possibly via an intelligent seatbelt[1]. This seatbelt can also be used to obtain the relevant *respiratory* measures. *Face videos* can be obtained using in-vehicle facing cameras. During manual driving standard positions of such videos can be used. However, it is possible that while driving in SB or TtS the orientation of the driver obstructs face detection from solely this camera angle. In that case, *additional cameras* observing the participant from different angles are needed.

Obtaining a measure for 'time to driver fitness' (TTDF) or time to driver unfitness' (TTDU) due to fatigue also requires prediction of the *future fatigue* level of the driver. Two approaches have been explored to this end. In the first and simplest approach, a table (Table B in Figure 3) is used to link each KSS score as estimated by the fatigue detection algorithm to an estimate for time to fitness and unfitness based on literature and expert opinions. In Table 1 an example of TTDU and TTDF estimations using the table-based approach to fatigue prediction is shown. The results from the on road and simulator fatigue studies described in D1.2 (Borowsky et al., 2020), show that sleep

[1] MEDIATOR partner Autoliv is developing such an intelligent steering wheel and seatbelt.

progression differs when driving manually or in CM. The TTDU/TTDF estimates should therefore depend on the current automation level. If the sleep progression differences are the result of the driving task related workload between the two scenarios, possibly also *(driving) context* should be taken into account when predicting the sleep progression.

Table 1: Example table for linking fatigue estimates to TTDU and TTDF (Table B in Figure 3).

KSS Score	KSS Meaning	TTDU	TTDF
1	extremely alert	2 hours	0
⋮	⋮	⋮	⋮
6	some signs of sleepiness	0.5 hours	0
⋮	⋮	⋮	⋮
9	very sleepy, great effort to keep awake, fighting sleep	0	0.5 hours

In the second approach, the fatigue detection algorithms are further developed to also include KSS score predictions based on historical data (Algorithm G in Figure 3). While this second approach has the potential to become much more accurate than the table-based approach, it is possible that the more feasible table-based approach is accurate enough for the current application in the MEDIATOR project.

For decisions related to improving driver fitness, information on the type of fatigue (sleep, underload or overload related fatigue) can be valuable. For example, if a driver is fatigued due to underload, providing additional workload might improve his fitness. If, however, the driver is fatigued due to overload or sleep deprivation, increasing workload is counterproductive. Additionally, understanding what caused the high fatigue level can improve accuracy of the fatigue prediction. If the specific fatigue type can be estimated, also the TTDU/TTDF estimates in Table 1 should be fatigue type dependent. There are two ideas on how to obtain such information that require further development in the next stage of the project.

- Idea 1: Implement a stepwise approach to determine fatigue type, where in the first step existing biomathematical fatigue models are used to determine the likelihood of the detected fatigue being sleep related. If this is not deemed likely, the driver’s workload could be estimated based on the driving context, driving mode and secondary task engagement. If a high workload is estimated, the measured fatigue level is likely overload related, while if low workload is estimated, underload is a more likely cause of the fatigue level.
- Idea 2: When fatigue is detected, a stimulus response task can be initiated. Analysis of the driver’s reaction to such task, in terms of reaction time and changes in for example pupil size, may be used to distinguish between fatigue types.

3.1.1.2. Distraction

Two *distraction detection* algorithms are being developed which estimate the situation awareness of the drivers. The first algorithm focusses on eyes off road detection (Algorithm C in Figure 3), while the second algorithm focusses on detecting which non-related driving task (NDRT) is being performed.

The *eyes off road*-based distraction detection algorithm can be used to assess the situation awareness of the driver. This algorithm will be best applicable while driving manually or in CM, as continuously maintaining situation awareness is expected of the driver in these cases. The algorithm first extracts relevant features from *face video* using *personalised* algorithms. These features are then used to determine if the driver is looking at the road or not. This gaze direction is then used to determine if the driver has sufficient situation awareness via the AttenD algorithm (Ahlstrom, 2013). The AttenD algorithm makes use of a time buffer that runs empty when the driver looks away from the road and fills while the driver looks on the road. The driver is assumed to be closer to a distracted state and thus have less situation awareness when the buffer is closer to zero. The eyes off road-based distraction detection algorithm calculates both the binary state of being distracted or not and a continuous distraction severity based on the duration in which the AttenD time buffer is zero. The preliminary eyes off road-based distraction detection algorithm designed for the MEDIATOR project showed promising results (Borowsky et al, 2020). However, further improvements are possible as described below:

- Currently, the time buffer has a maximum of 2 seconds to indicate that full situation awareness is again obtained. However, it is possible that this buffer, or the way in which this buffer is filled and depleted depends on the *driving context*. For example, when driving in a complex urban environment, the driver might need more time to regain full situation awareness compared to driving in a far less complicated highway environment. In the next stage of the project the aim is therefore to make the algorithm context dependent.
- The current algorithm only detects eyes on or off road, while the original AttenD algorithm also takes account of other driving related eye gazes such as those directed at mirrors or side windows. In the next stage of the project the aim is therefore to include more specific eye gaze directions to make full use of the original AttenD algorithm.
- The adopted distraction severity definition might be further explored and correlated with effects on situation awareness and driver performance. One proposition is to not only look at the duration of the time buffer at zero, but also take into account lower levels of distraction where the time buffer has not yet reached zero. Due to the possibly high frequent nature of the distraction severity measure, some filtering might be needed before using it for the TTDU calculation.

Correctly predicting driver distraction in the near future is very complex as it depends on many (often unmeasurable) factors, such as driver intention. However, the estimation of TTDU due to distraction can instead be based on the expected loss of situation awareness due to distraction. Similar as it was proposed for fatigue related TTDU (but with a much shorter time scale), the distraction *severity* can be linked to the expected time to driver unfitness. In Table 2 an example of such a table is shown (corresponds to Table C in Figure 3). The final contents of this table will mainly be based on literature and expert opinions in the next stage of the project.

Table 2: Example table for linking distraction severity to TTDU during driving manually or in CM (Table C in Figure 3).

Distraction severity level	Meaning	Calculation	TTDU
1	Sufficient SA	AttenD Time buffer close to max	Order of minutes
2	Slightly reduced SA	AttenD time buffer close to zero	Order of 10 seconds
3	Short term loss of SA	AttenD time buffer zero for short period	Order of 1-5 seconds
4	Long term loss of SA	AttenD time buffer zero for long period	0

The second algorithm focusses on detecting which *non-related driving task* (NDRT) the driver is performing. While driving in SB or TtS, the driver is not expected to maintain a high level of situation awareness all the time. Instead, the driver is expected to be fit enough for the current automation functioning. This means that the time to driver fitness (TTDF) should always be smaller than the time to automation unfitness (TTAU). Research has shown that NDRTs can affect the required takeover time in varying degrees, e.g., task modality influences takeover time (Wandtner 2018) and NDRTs causing motoric workload increase takeover time (Naujoks, 2019). Detecting which NDRT is being performed can therefore aid in estimating the TTDF with respect to distraction.

The algorithm is based on two separate approaches. A first approach uses wide angle and face cameras to determine secondary task involvement and task type (Algorithm D in Figure 3). This algorithm will, for example, determine which in-vehicle devices are being used. The second approach instead focusses on eye glance patterns obtained from face videos to determine secondary task type and how much attention is still paid to the road (Algorithm E in Figure 3). To obtain an estimate of TTDF, the NDRT type should be linked to average takeover times when performing these NDRTs found in literature. An example of such a table is shown in Table 3 (corresponds to Table D in Figure 3).

Table 3: Example table linking NDRT type to TTDF (Table D in Figure 3)

NDRT type	TTDF	NDRT examples
Messaging	5 seconds	Short messages via phone or tablet where driver intermittently looks at the road
Obstruction	10 seconds	Working on a laptop
Immersion	10 seconds	Watching a movie on a fixed screen
Obstruction and immersion	20 seconds	Watching a movie on a laptop

In D1.2 (Borowski et al., 2020) a simulator experiment is described, in which driver's hazard perception skills were measured while driving manually or in CM and with or without performing a secondary task. Results from this experiment can aid with filling both Table 2 and Table 3, as it describes the relation between the fitness related measure "hazard perception" and different levels of distraction.

3.1.1.3. Time to driver fitness and unfitness

The overall time to driver fitness and unfitness are a combination of Table 1, Table 2 and Table 3. The *shortest* time to driver unfitness and *longest* time to driver fitness are selected and serve as output to the decision logic. Both variables will also require a *worst, likely* and *best* case of that value, which is deduced from the *reliability* of the estimates.

3.1.2. Time to driver discomfort

Next to the driver fitness, also the driver comfort is estimated, and actions can be taken to improve driver comfort. The main focus in the MEDIATOR project related to comfort is to initiate takeovers. Other actions such as adjusting automation settings could also improve comfort but are outside of the scope of the MEDIATOR project. Two approaches are being taken to *detect* and *predict driver comfort*. The first approach is an offline approach where typical driving situations are assigned average comfort values. The second approach is based on real-time comfort detection from face video data. Depending on the feasibility of the second approach, either one or both approaches can be used to determine time to driver discomfort (TTDD).

3.1.2.1. Offline comfort detection

For the offline comfort detection approach Table 12 in D1.2 (Borowsky et al., 2020) with potentially uncomfortable driving situations is established based on literature (first column corresponds to Table A and first and last column correspond to Table F in Figure 3). This table covers two types of uncomfortable situations: 1) potentially uncomfortable situations in manual driving and 2) potentially uncomfortable situations in automated driving. The table contains a brief description of the situation, the probability of a decrease in driver comfort, a suggestion to adjust this probability based on driver competences, the probability that a situation can be detected in advance, and the time span within which this situation can be detected in advance. A summary of this table is shown here in Table 4, the original table with more detailed driving situations can be found in D1.2 (Borowsky et al., 2020).

Table 4: Summary of Table 12 in D1.2 (Borowsky et al., 2020), describing potentially uncomfortable driving situations during either manual or automated driving, the expected probability for a decrease in comfort, the probability to detect the situations and the possible time span for detection in advance (combination of Table A and Table F in Figure 3).

Situation		Average probability of a decrease in drivers' comfort		Average probability to detect situations in advance	Average time span for the premature detection of the situations
		Starting value	Suggested adaptions		
Manual driving	Car following scenarios	55%	increase for inexperienced drivers	high	several minutes
	Situations with poor visibility	45%	increase for elderly drivers	high	several minutes
	Situations with high complexity	40%	increase for elderly drivers	high	several seconds or minutes
	Challenging driving manoeuvres	29%	increase for elderly drivers	medium to high	several seconds
	Situations causing high levels of uncertainty	50%	increase for elderly drivers	high	several minutes
	Demanding or highly prioritized NDRTs	35%	increase for younger drivers	medium	several seconds
	Impaired driver state	57%	/	high	several seconds or minutes
	Longer and monotonous trips	45%	/	high	several seconds or minutes

	Optimise fuel or energy efficiency	80%	/	high	several seconds or minutes
Automated driving	Situations with higher risks of motion sickness	62.5%	/	medium	several seconds
	Situations with high demands on communication with other road users	50%	/	high	several seconds or minutes
	Driving under time pressure	50%	/	medium	several seconds or minutes
	Purpose of the trip (e.g., driving for pleasure)	50%	/	high	several seconds or minutes
	Situations that cannot be handled by the automated system	100%	/	high	several seconds or minutes

3.1.2.2. Real time comfort detection

The second approach focusses on real time detection of driver comfort from face video data (Algorithm F in Figure 3). Unlike the offline approach, which is focused on predicting driver discomfort, this approach is most valuable for detecting *current* driver discomfort. As the real time approach is assessing comfort of an individual driver, it is likely more accurate for this specific driver than the offline method based on group averages in general situations.

This approach, however, is still in its research phase and initial results were described in D1.2 by Borowsky et al. (2020). Facial recognition software was used to extract action units, i.e., movements of specific individual facial muscles or muscle groups that reflect distinct momentary changes in facial appearance, during a close approach manoeuvre. Such manoeuvres are known to cause discomfort (Beggiato et al., 2019). Results on face tracking show that any deviations from a central frontal perspective of the driver's face or obstruction of face parts, such as reflecting eyeglasses and beards, significantly reduces face tracking quality. This should be taken into account when installing face cameras for face tracking purposes. Results on the comfort estimation studies show that action units related to eyes kept open, eye blinks, raises of upper eye lids and inner brows, lips being pressed and stretched and an upward oriented lip pressing motion can point towards a reaction of surprise, tension and visual attention during the close approach situation. Thus, the comfort detection algorithm that will be developed should take these action units into account. Finally, the results show relevant differences between drivers, such as the combination of driver characteristics showing a higher extent of extraversion, sensation seeking tendencies, uncertainty tolerance, desirability of control and low age being correlated with a high effect on action unit changes.

There are two main applications for the current driver discomfort estimates. First of all, corrective actions can be initiated when driver discomfort is detected. Secondly, the probability of discomfort in the table developed in the offline approach can be updated with the real time comfort measures of these situations. It might even be possible to identify new uncomfortable situations for that specific driver (personalisation) and add them to the table.

3.1.2.3. Time to driver discomfort

The *time to driver discomfort* can be derived from *context information* and the table developed in the offline comfort detection approach. While each situation in Table 4 has a different *probability of being uncomfortable*, they all are potentially uncomfortable and should thus evoke a suggestion from the Mediator system for taking over control. This means that the time to driver discomfort can simply be calculated as the *time until such an event is expected* to take place.

3.1.3. Intervention type

To determine which action is most suited to improve driver fitness in each situation, the decision logic needs to have an *estimate of the improvement* that such intervention will lead to and a *duration* of when this result is reached. Additionally, in order to monitor the progress and act upon deviations from the expected progress, an estimate of *the expected progress* needs to be known to the decision logic. Such information can be summarised as in Table 5 (corresponds to Table E in Figure 3) and read out by the decision logic module.

Table 5: Fitness intervention information from Driver module (Table E in Figure 3)

Action	Expected fitness improvement	Progress
1	X [sec]	F(t)
2	Y [sec]	G(t)
…	…	…

The progress functions will probably be based on simple standard functions, such as a linear function, parametrized using only the expected fitness improvement and the duration until the result is reached.

The actual effectiveness of the intervention type to improve driver fitness depends strongly on how these actions are implemented in the HMI. As at this stage the exact implementation is not yet known, it is not possible to fill in Table 5 just yet. However, once the implementations are known, experts on fatigue and distraction within the MEDIATOR consortium will be consulted to complete the table.

The interventions to improve comfort differ somewhat from the interventions to improve driver fitness. Within the MEDIATOR project, all comfort improvements entail switching control between driver and automation. Furthermore, comfort interventions are always optional and will not be initiated if they cause safety hazards. Next to the time to driver discomfort, the decision logic thus only requires *probability* that comfort will be increased as a consequence of the transfer of control, which can be found in Table 4. The comfort interventions corresponding to a finite time to discomfort will therefore resemble the example in Table 6 (corresponds to Table G in Figure 3).

Table 6: Comfort intervention information from Driver module (Table G in Figure 3)

Action (Control transfer)	Probability of comfort improvement
Manual to SB/TtS	W
SB/TtS to manual	X
Switch on CM	Y
Switch off CM	Z

3.1.4. Driver state class

The third output of the Driver module is the actual *driver state class*, which provides the type or reason for degraded fitness or comfort. This state class has two entries, one related to driver fitness and one related to driver comfort. The driver fitness entry can take on one of the values “normal”, “fatigued” or “distracted”. Depending on the feasibility of distinguishing between sleep, underload or overload related fatigue, the value “fatigue” can include references to the specific fatigue type. The driver comfort entry can either take on the value "comfortable" or "uncomfortable" with the identified *reason* for this discomfort.

3.1.5. Personalisation

The driver state estimation will be personalised in two ways: using *personalised algorithms* for fitness detection and *personalised comfort estimates*. The fatigue, distraction and real-time comfort detection algorithms make use of *driver IDs* to tune their driver specific models. Such an ID can be extracted from face video data using facial recognition software. The probability of discomfort in the offline comfort table can be adjusted based on *historical data* regarding the acceptance or rejection of Mediator system suggestions to transfer control for each situation. Additionally, driver information such as *age* can be used to already tailor the baseline values of this probability. These personalisation features need to adhere to privacy regulations. For example, informed consent should be given by the driver prior to activation of such features.

3.1.6. Overview

Figure 3 provides an overview of the inputs, algorithms and tables described in the previous paragraphs. Each algorithm or table refers to one of the algorithms or tables discussed in the previous paragraphs. Two of the algorithms are shown in grey, which is to indicate that it is uncertain if they can be used in the final Mediator system as they are still in an early research phase. The inputs required for all these algorithms are shown on the left and divided in hardware and data from other modules. The modules from where the data is expected to come is indicated in grey with "C" for the context module, "AU" for the automation module and "DL" for the decision logic module. The outputs of the Driver module are shown on the right of the figure. Time to driver fitness/unfitness and driver state class follow from "merge" block within the TTD block. The time to driver discomfort and the corresponding comfort class follow from the block "Comfort to TTDD". Finally, the fitness and comfort interventions and their expected impact are outputs of the block "Interventions".

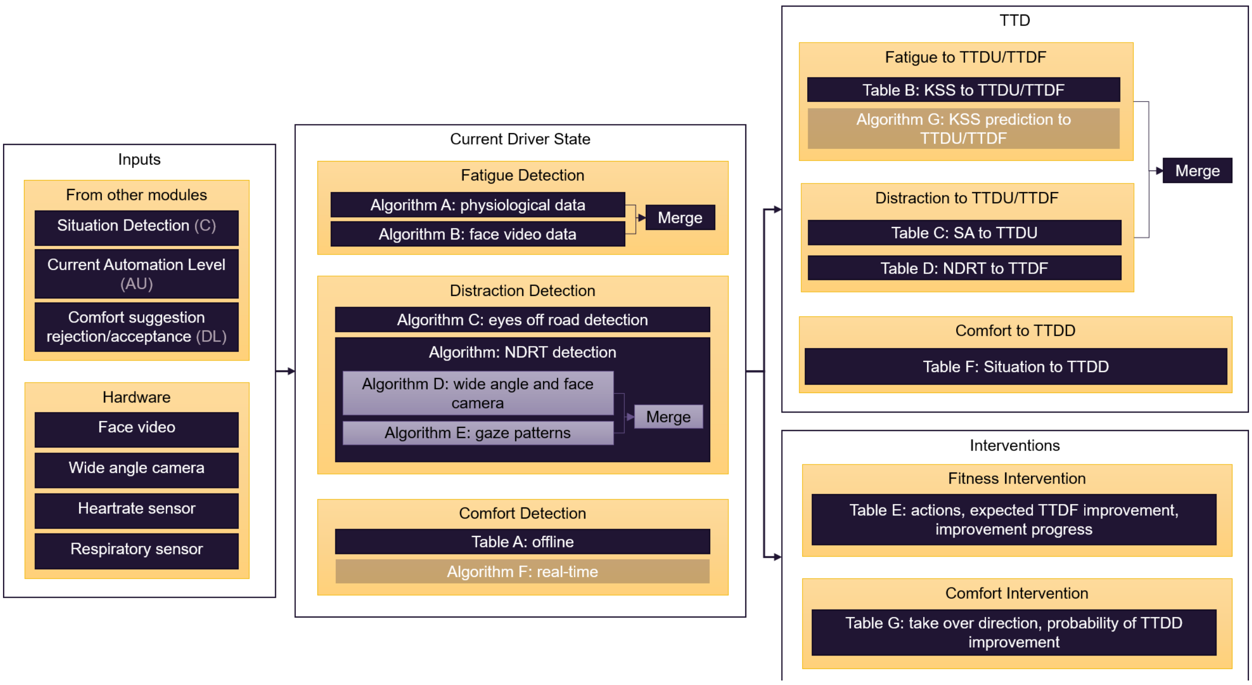


Figure 3: Overview Driver Module

3.1.7. Key performance indicators

The Driver module outputs time to driver fitness and unfitness, expected outcomes of interventions and the current driver state class. To assess the correctness of these outputs the following key performance indicators can be used.

- True/False positive/negatives of the detected driver state and derived measures from them, such as the area under the curve (AUC), sensitivity and accuracy
  - Receiver operating characteristic (ROC) curves can be used to visualise some of these results
  - Ground truth measures are difficult to obtain. Subjective measures, such as KSS fatigue scores and questionnaire results, will most likely be used.
  - Thresholds for fitness/unfitness can be based on literature and/or assessed via driving performance measures such as time to collision and lane deviation.
- Root mean square error (or similar measure) of time to driver fitness/unfitness/discomfort.
- Root mean square error (or similar measure) of the achieved fitness improvement.

3.1.8. Functional Requirements

The following software-related functional requirements for the Driver module were identified. These requirements largely result from the theoretical elaboration of the options for the Driver module as well as the practical feasibility of these options as presented in the previous paragraphs.
The driver module should:

- Estimate the time to driver (un)fitness for the best, likely and worst-case scenarios for the current driving context
  - Detect the current driver fitness
    - Detect the current level of driver fatigue
      - Detect heart rate, respiratory and facial features
      - Estimate KSS score
    - Detect the current level of driver distraction
      - Detect eyes on/off road
      - Estimate distraction severity based on eyes off/on road
      - Detect which non-related driving task (NDRT) is performed
  - Predict when driver unfitness will be reached
    - Predict when the driver distraction will reach the level of degraded fitness
      - Predict loss of situation awareness based on distraction severity
      - Predict time to driver fitness based on NDRT involvement
    - Predict when the driver fatigue will reach the level of degraded fitness
      - Predict fatigue progression based on current KSS score
  - Estimate time to driver fitness as the longest estimate from distraction or fatigue estimates
  - Estimate time to driver unfitness as the shortest estimate from distraction or fatigue estimates
  - Estimate worst, likely and best cases using reliability of the underlying estimates
  - Request context relevant information from the context module
    - Possible information can be time of day and situation complexity
- Estimate the time to driver discomfort for the best, likely and worst-case scenarios
  - The system shall compare upcoming driving situations with the identified uncomfortable driving situations
    - The system shall request relevant information from the context module
  - Estimate the time until the uncomfortable situation will occur

  - Estimate worst, likely and best-case scenarios using the probability that comfort will be increased
- Predict the impact of fitness improvement interventions on the driver fitness
  - Predict the final impact improvement and provide corresponding accuracy
  - Predict the duration until the improvement is achieved
  - Predict the improvement progress over time and provide corresponding accuracy
- Predict the impact of comfort improvement takeovers
  - Estimate the likelihood that a takeover will improve driver comfort
- Estimate the drive state class
  - Estimate which driver state is responsible for the expected degraded fitness
    - Values can be none, distracted, fatigued.
  - Estimate what is the reason for expected discomfort
- Personalise driver state estimations (if informed consent is given)
  - Detect driver ID
    - Possibly request sleep/driving history and driver age via HMI
  - Estimate driver fitness with individualised algorithms tuned to specific drivers
  - Estimate comfort based on historical data on user acceptance or rejection of takeover suggestions

Additionally, some corresponding hardware-related functional requirements were identified. The system should:

- measure heart rate
- measure respiratory rate
- capture face video data from different angles
  - facial features should be visible while driving and while performing NDRT
- capture wide angle video data
  - type of NDRT involvement should be visible from video

3.2. Automation Module

The automation module is responsible for the interface between the Mediator system and the vehicle automation functions. Unlike the driver module, which is mainly responsible for driver monitoring, this automation module is also responsible for the communication to the vehicle automation and retrieving additional information available from the automation for use in other modules. This module therefore has the following three main functions:

- Provide automation state information,
- Adjust the automation state,
- Provide data for the context module.

Apart from defining these functions, a challenge for the automation module is to retrieve the required input for these functions from different types of vehicles. Such *standardisation* requires detailed knowledge of automation software and hardware and their accessibility for a range of other vehicles.

In the following sections the main functions of the automation module are briefly described. The section is concluded with an overview of key performance indicators and functional requirements for the automation module.

In contrast to the driver module, which has known degraded performance markers such as fatigue or distraction as well as methods to quantify them, there are no established methods for assessing automation fitness. This section provides the foundations and concepts detailed in the Mediator deliverable of Task 1.3 (Mano et al., 2021) to allow the automation state module to estimate the automation fitness for any driving automation system. The focus of the work for the automation state module lies in developing and evaluating these concepts within the MEDIATOR project.

Finally, it must be noted that the role of the automation state module is not to improve the performance of the driving automation system, but to predict when it will not perform well. The automation state module interfaces with the driving automation system but does not alter how it behaves in any ways.

3.2.1. Automation State

The automation state includes the time to automation (un)fitness, possible intervention type, automation state class and current automation level in use. In the following paragraphs, these variables are described and initial ideas for obtaining them from in-vehicle available data are proposed.

3.2.1.1. Time to automation (un)fitness

The time to automation (un)fitness describes how much time is left before an automation level will no longer be available (unfitness) or become available (fitness). It is the most important variable outputted by the automation module as it provides the decision logic with the means to trade off driver and automation fitness using comparable variables and taking future situations into account. The latter ensures a preventive approach to safety critical situations rather than a reactive one.

The time to automation (un)fitness should be calculated *continuously* for *each* automation functionality, which in the MEDIATOR project refers to automation levels CM, SB and TtS.
The time to automation unfitness (TTAU) refers to the time up till which the automation function performance can be guaranteed with a certain level of certainty. Within the MEDIATOR project, to quantify the automation fitness, an automation fitness scale is introduced, which corresponds to the rate of automation system deactivations, overrides or fallbacks (i.e., the number of occurrences of system deactivations/overrides/fallbacks per X hours). The higher on the scale, the less frequent are the system deactivations and so the more fit the automation as illustrated in Figure 4 below.

5 — Maximum performance
Rare occurrences of system deactivations/overrides/fallbacks per X hours

…

1 — No performance
Frequent occurrences of system deactivations/overrides/fallbacks per X hours

Figure 4: Automation fitness scale representation

Using collected and enriched (by annotation) data, the goal is to correlate both the automation indicators and the driving context (see Figure 5 below) with the number of occurrences of system deactivations/overrides/fallback initiations per time unit normalized on the automation fitness scale.

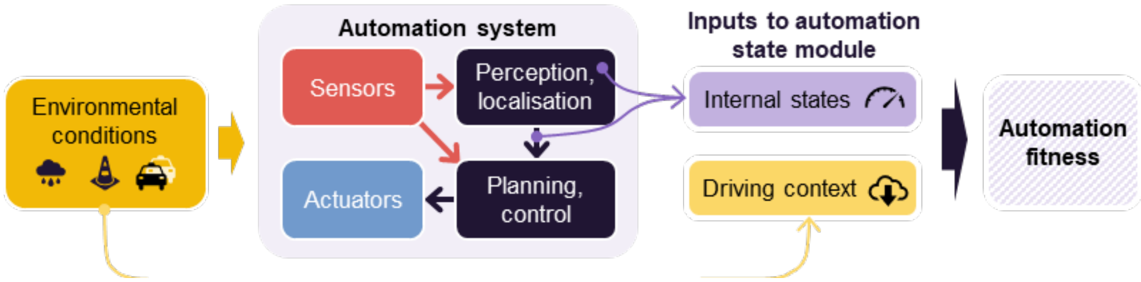


Figure 5: Simplified system architecture and inputs of the automation state module

The outcome of the method is an estimation of an automation fitness (AF) score on the scale using both online observations of the automation indicators as well as online observations and predictions of the driving context. The estimation of the AF score is then used to predict the time to automation (un)fitness using cut-off thresholds; for instance, TTAU would be the shortest time when the estimated AF score becomes lower than a cut-off threshold.

As an example, in Figure 6, the AF score (in green) is estimated in a near future using the predicted driving context which forecasts a limited increased traffic density followed later on by a section of roadwork (black icons). The AF score is estimated to decrease during these two sections based on the correlation built with the prior data collection. TTAU can then be estimated as the shortest time when the predicted AF score becomes lower than a predefined cut-off threshold (in red), which will happen when reaching the roadwork.

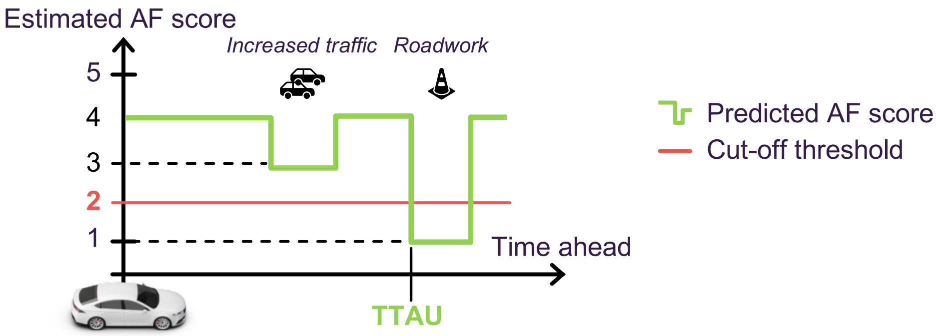


Figure 6: TTAU prediction using automation fitness score

There are a few exceptions to this method:

- **Worst case TTAU in CM**: Given that any driving automation system up to and including SAE level 2 gives no guarantee on positive system performance (it will always rely on the driver to supervise the system) the worst case TTAU that can be expected in CM mode will always be 0 seconds,
- **In SB and TtS**: In SAE level 4 driving automation systems, the system is by definition always fit to drive within its Operation Design Domain (ODD), both for handling the Dynamic Driving Task (DDT), the monitoring and the fallback. This means that, if SAE level 4 is used during SB or TtS, the TTAU would be infinite. Practically however, a driver will prefer to take over from the system before a fallback happens, for example at the end of the ODD. Therefore, it might be desired to adapt the definition of TTAU in such cases to reflect the time until fallback rather than time to automation unfitness.

Three estimations with different certainty levels are requested: worst case, likely case, best case. In Figure 7 below, we illustrate the estimation of TTAU for the **CM level**. In this example, it is assumed that the automation state module receives the information that a dense traffic is predicted ahead for a short duration (represented with the three black vehicles) followed by a roadwork later on. The TTAU predictions are built as follows:

- **TTAU worst case** is always equal to zero seconds in the CM level (independently of the current or predicted AF score estimation),
- **TTAU likely case** could consider the traffic density prediction “as is”, i.e. taking into account the dense traffic in the middle part,
- **TTAU best case** could consider a more favourable (or optimistic) traffic prediction, such as light or moderate traffic density, which would lead to a lower variation of the AF score.

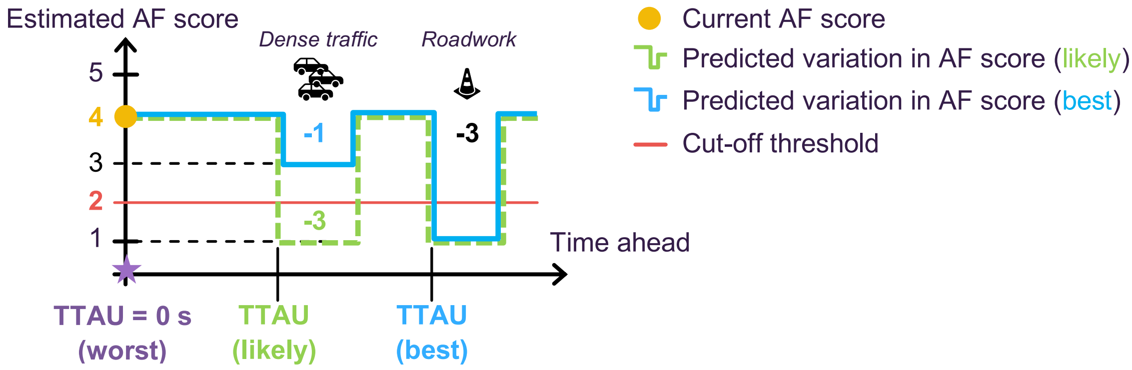


Figure 7: TTAU prediction for continuous mediation level (CM)

Table 7 summarises the TTAU estimations for the worst, likely and best-case scenarios and for all three Mediator automation levels.

Table 7: TTAU estimations for the worst, likely and best-case scenarios

TTAU estimation	Certainty level	Underlying assumptions	Supervised (CM)	Unsupervised (SB, TtS)
Worst case	High	Conservative	0 s [(1)]	Function of AF score
Likely case	Medium	Pragmatic	Function of AF score	
Best case	Low	Optimistic	Function of AF score	

[(1)] As stated above in the same section, TTAU estimations in the worst case for CM is always considered to be zero second as the driver is always in the loop.

3.2.1.2. Automation state class

The automation state class variable describes the reasons for upcoming automation fitness or unfitness. To compute the automation state class, it might be possible to identify which piece of the driving context information weighs the heaviest in a variation of the automation fitness score (using knowledge on the correlation built with the data collection). In the simple example in Figure 7, we assume that the driving context is composed of weather conditions and roadwork locations. As there is a roadwork location ahead and the weather is unchanged, the decrease in automation fitness score estimation from 4 to 1 is therefore linked to the roadwork in the system logic.

Cases might not always be so clear-cut though. It is likely that some roadworks will not constitute a problem for the DAS in good visibility, but at night, in poor visibility, or without a lead vehicle they might. The automation state class in this case might therefore need to be something more abstract such as the probability of “bad perception”, “bad lane markings” and “roadworks”.

In any more complex case where there has been no previous decision of what external context constitutes the issue for the DAS, the automation state class will probably become “end of ODD”.

The automation state class outputs and timings must be developed further for improved messaging to the user, and in collaboration with the HMI component and decision-making component of the Mediator system. This variable is important for the HMI to, for example, indicate to the driver why control will need to be taken over as to increase system transparency and thus improve the driver's understanding of the system. Automation state class can also be used to determine possible actions, such as rerouting to avoid leaving the highway and therefore staying within the automation's ODD, but, as indicated in the next section, that is beyond the scope of the current project.

3.2.1.3. Intervention type

As the automation state module knows when (TTAU/F) and why (automation state class) there is an upcoming change in automation fitness, it can propose possible actions it thinks are suitable to take to improve the automation fitness.

As stated in Section 4.3.3 in MEDIATOR deliverable of Task 1.1 (Christoph et al., 2019), the Mediator system can propose to activate or deactivate the DAS depending on whether the system is becoming fit or unfit to drive.

In the case of the system becoming unfit, other alternative actions could possibly increase the time during which the automation remains fit (i.e., TTAU). In the example above about the roadwork, these actions an others could be appropriate:

- "Reduce speed" to increase the time to reach the roadwork location,
- "Change lane" if the roadwork is known to only affect the current lane,
- "Take a different route" to avoid the roadwork, having checked beforehand that the alternate route leads to a prolonged fitness, comparatively.

However, such interventions are beyond the scope of the MEDIATOR project. For further development of the Mediator system, however, it can be helpful to already take the possibility of *improving automation fitness* into account in the design of the current system.

3.2.1.4. Current automation level

A simple but very relevant output of the automation module is the *current automation level* that is active. This variable can take the value of none, CM, SB or TtS.

3.2.2. Adjusting automation state

In current vehicles it is not possible to *adjust the automation state* other than via user input. However, with the eye on an integrated HMI that is easy to understand for drivers, it might be beneficial to let the Mediator system have access to the automation and send automation change requests. This does, however, bring a new level of responsibility to the Mediator system, which should be taken into account when assessing usability of the system from an OEM point of view.

3.2.3. Context relevant information

The third function of the automation module is to *extract context relevant information* from the vehicle automation software for use in the Mediator system. Such information can generally be split up in three categories: on-board sensors and processing, third party information sources, and in-vehicle entertainment information. Table 8 provides some examples of context relevant information which could be extracted from the automation. Within each category both raw sensor data and data already (partly) processed by the automation software could be requested.

Table 8: Examples of context relevant information possibly required from the automation module

Information type	Examples

On-board sensors	Road facing camera RADAR/LIDAR Detected objects in surrounding traffic GPS Kinematic data (CAN bus data) Estimated sensor reliability
Third party	Live traffic data and forecasting, e.g., traffic jam HD map information e.g., road types, upcoming on/off-ramps, roundabouts and intersections
In-vehicle entertainment	Infotainment usage Connected smart phone status

3.2.4. Key performance indicators

The automation state module outputs time to automation fitness and unfitness, automation state class, current automation level and relevant context information. To assess the correctness of these outputs, key performance indicators (KPI) will be used.

For instance, time to automation (un)fitness could be described as a classification problem on whether the automation state module has correctly predicted a change in automation fitness versus the observation therefore involving standard metrics based on true/false positive/negative. Moreover, assuming that a change in automation fitness was correctly predicted by the automation state module at a certain location in the planned route, it could also be of interest to evaluate how good was the time prediction before reaching the location compared to the time it actually took. Indeed, the estimated time to reach a certain location is dependent on multiple factors such as the traffic speed provided by an online source. Making such estimated time assessment therefore involves to also assess the accuracy of the various pieces of context information used to build the driving context.

Since the methodology to estimate the automation fitness is new, the KPIs will be developed in more details during/after the data collection part of the development of the automation state module. The analysis of the collected data to build the correlations between the automation internal states, driving context and the automation fitness (from which the outputs of the automation state module are derived) will provide insights that will drive the definition KPIs.

3.2.5. Functional requirements

At this stage of the project the following functional requirements were identified for the automation module. The automation state module should:

- Estimate worst, likely and best-case time to automation (un)fitness based on automation fitness estimates for the current driving context
    - Estimate the current automation fitness
        - Estimate the current automation fitness score
        - Access relevant information from the automation system to estimate the current automation fitness score
    - Estimate the predicted automation fitness
        - Estimate the predicted automation fitness score
        - Access relevant external driving context information
    - Estimate when the driving automation system is unfit to drive

        - The driving automation system is deemed unfit to drive if it can no longer execute its defined dynamic driving task due to degraded automation performance (low automation fitness score)
    - Estimate the time to automation unfitness as the shortest time when the estimated automation fitness score becomes lower than a cut-off threshold
    - Estimate the time to automation fitness as the shortest time when the estimated automation fitness score becomes greater than a cut-off threshold
    - Estimate worst, likely and best-case scenarios of time to automation (un)fitness using the reliability of its inputs, both internal and external
- Determine the active automation level as either none, CM, SB or TtS
- Determine the automation state class, i.e., the reason for an upcoming change in automation availability
- Determine the appropriate intervention type, i.e., a possible way to improve the automation fitness
- Extract and collect context relevant information from the driving automation system to the context module

Additionally, a set of recommendations are proposed for the development, testing and evaluation of the automation state module:

- Drivers should comply with the speed limit or lower
- The automation state module should know the planned route

3.3. Context Module

When deciding who is fittest to drive and how to improve the driver comfort, the Mediator system also takes account of the driving context. To achieve this, a context module will be developed that retrieves and integrates context relevant information from different sources, such as the automation module, the driver state in-vehicle cameras and additional context specific hardware. The context module will combine this information into specific context characteristics requested by other modules, mainly the driver, HMI and decision logic modules. A complete overview of context information needed by these modules will be developed in the next stage of the project. However, it is clear that the driver module will require context information to identify the uncomfortable situations described in Table 4 as well as in-vehicle and outside context information to determine the driver workload. The decision logic module will need information on, for example, route information to optimise for upcoming traffic situations.

In this section, a brief initial overview of typical information which can be outputted by the context module is provided in Table 9. Such overview helps to specify exactly which type of information is sufficient for algorithms of the other modules. The list is not exhaustive and leaves room for specific needs that may arise in the next stages of the project.

The table provides a short description of the available information type in the first column, the corresponding unit in the second column, and subsequently a description over which time frame this information is available, split in current, near future (generally order of minutes) or over the entire route in columns three, four and five, respectively.

Table 9: Initial list of context information outputted by the context module

Variable	Value/unit	Current	Near Future	Route

Road marking quality	Quality: high, medium, low	Yes, based on on-board vehicle sensors, road facing camera	No	No
Road type	Highway, urban, city	Yes, GPS + map matching	Yes, based on active route in navigation systems	Yes, based on active route in navigation systems
Speed Limit	Number in km/h	Yes, based on detection of traffic signs, V2I information and GPS + map matching	Yes, V2I information and GPS + map matching	Yes, V2I information and GPS + map matching
Infrastructure type	Separate, shared with VRU	Yes, GPS + map matching	Yes, based on active route in navigation systems	Yes, based on active route in navigation systems
Traffic jam	yes/no + upcoming traffic jams in km	Yes, based on traffic data and on-board vehicle sensors	Yes, based on traffic predictions from third parties	Yes, based on traffic predictions from third parties >> but low certainty
Events ahead	Accidents, road works	Yes, based on online services if available	Yes, based on online services if available	Yes, based on online services if available
Traffic density	Vehicles in current scene	Yes, based on traffic data and on-board vehicle sensors	Yes, based on traffic predictions from third parties, perhaps some simple predictive modelling	Yes, based on traffic predictions from third parties >> but low certainty
Traffic speed	Float value	Yes, based on third party floating car data / web services if available (like RDW in NL)	Yes, downstream information available, based on third party floating car data / web services if available (like RDW in NL), perhaps some simple predictions	Yes, based on third party floating car data / web services if available (like RDW in NL)
VRU density	VRU count in current scene	Yes, based on traffic data and on-board context sensors	Yes, based on road types from high-definition map	Yes, based on road types from high-definition map >> but low certainty
Weather	rain/wind/snow /ice/etc	Yes, based on on-board context specific sensors and third-party data	Yes, based on weather forecast from third party	Yes, based on weather forecast from third party

The vehicle on-board sensors mentioned in Table 9 refer to sensors that are standard in the vehicle, also without Mediator system. If this information is indeed used by other modules, a requirement for any vehicle using the Mediator system would thus become that the on-board sensors of that vehicle can output such information. Table 9 also mentions context specific on-board sensors, which refer to sensors that will be implemented with the Mediator system but are not yet mentioned in the hardware requirements of the other modules. Again, these systems only need to be implemented if the other modules indeed require this type of context information. Finally, the third-party data mentioned in Table 9 refers to data such as weather and traffic forecasts that do not require real sensors but might require other hardware to connect to the third-party services. It should also be noted that certain information, such as high-definition maps, is not always available. This means that, if one or more of the other modules require this information, an alternative robust solution for obtaining this information should be found.

3.4. HMI Module

To communicate between the vehicle related systems, including the Mediator system, and the human driver a dedicated Human-Machine Interface (HMI) will be designed. An HMI is defined as a device that enables humans to engage and interact with machines. In order to sufficiently consider the vast complexity of an HMI for partially autonomous vehicles while at the same time securing (intuitive) usability i.e., simplicity for humans, the development of the Mediator‘s HMI is based on a holistic approach (Christoph, et al., 2019). The Mediator HMI, which is being designed in a holistic approach, facilitates and manages all interaction between human and vehicle for both primary, driving related tasks as well as for secondary tasks (e.g., climate, entertainment). For example, as part of the holistic approach, rather than competing with existing HMI elements for driver attention, such as entertainment systems, we want to control their dominance. Also, existing HMI elements can be used within Mediator system functions, e.g., maintaining or eliciting driver fitness by changing climate control functions such as temperature or airflow.

The main HMI functions through this holistic approach are therefore:

- Supporting conventional driving tasks,
- Facilitating negotiations between driver and automation,
- Guiding control transfers between driver and automation (takeovers),
- Informing on CM switch off/switch on,
- Executing corrective actions to increase driver fitness,
- Executing preventive measures to maintain driver fitness,
- Detecting and facilitating driver preferences and momentary inputs

The HMI should also adhere to several non-functional requirements such as having high usability and transparency for its users and improving driver comfort and safety.

3.4.1. Conventional driving tasks

While this chapter focuses on the requirements of the Mediator HMI, it should also take account the conventional HMI requirements of (automated) vehicles in general, like steering, (emergency) breaking or control of the external HMI. Such an integrated approach ensures that the driver is optimally supported and has a comfortable and safe driving experience. The integrated HMI should therefore be designed for learned and familiar *affordances*, such that conventional HMI communication remains at least equally understandable and new functions are easily learned.

3.4.2. Negotiation Routine

The decision logic can request takeovers from human to automation or vice versa or switching CM on or off. With each takeover or CM switch on/off request, a level of necessity and a timeline are provided, which together indicate the level of urgency. Offering the *driver autonomy* over automation level decisions when possible, i.e., when necessity is medium to low, is crucial for *trust* and *user acceptance*. The way in which a request is communicated to the driver and how the actual ritual is executed, therefore differs depending on *necessity level*. In case the driver indicates a different preference than the decision logic’s (DL) preferred automation level, the HMI must negotiate with the driver. For low necessity levels a seductive negotiation between automation and human is applied, while for higher levels a persuasive negotiation is applied, or even a forced takeover (no negotiation) or switching on/off CM is applied. An overview of this negotiation routine is shown in Figure 8.

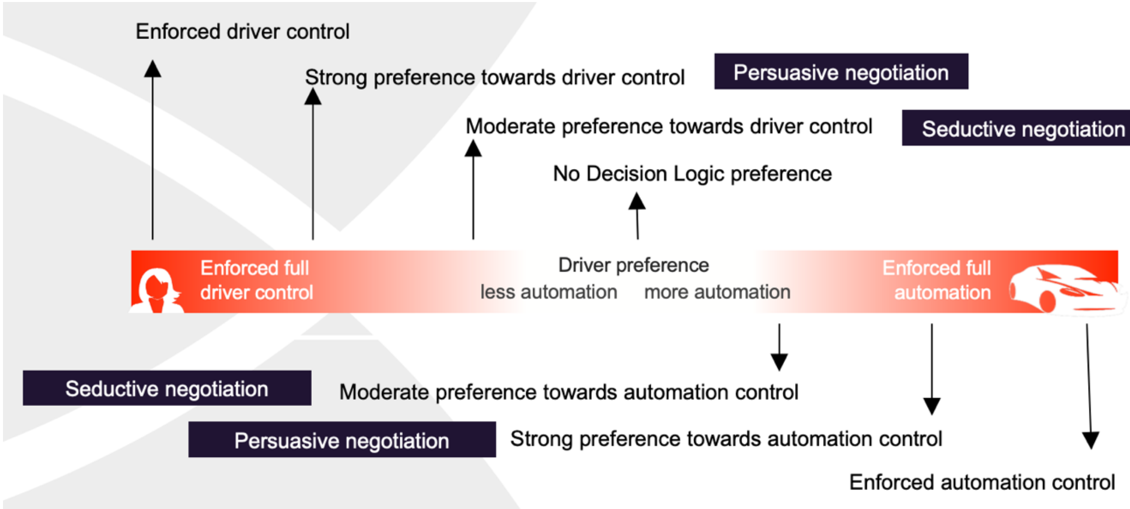


Figure 8: Takeover negotiation routine

3.4.3. Takeover procedure

The takeover procedure will follow a standard process (ritual) in which the driver is timely informed on the intended takeover, including measures to increase driver fitness if appropriate. The control transfer ritual, Figure 9, foresees signals at specific time intervals, and required driver responses. Time intervals, the number, multi-modality and intensity of signals are all variable, depending on time budget and driver response, as indicated by Decision Logic.

Figure 9: Control Transfer ritual with Signals, driver response time (Processing and Action) at time intervals

First, the driver is *informed* on the takeover making use of the *negotiation routine*. During this state the *necessity* of and the *reason* for the takeover are explained to the driver. The communication should be designed such that the stimuli are perceived by the driver *independently of the activity* they are performing. For example, if a driver is using his or her phone, displaying a takeover request on the dashboard display would not suffice. The HMI should therefore either communicate through the device which has the driver's attention, such as a mobile phone, or in such a way that it can be perceived in the driver's peripheral view or other non-visual senses. An example of the latter is to adjust the in-vehicle ambience using ambient lighting and possibly even changes in temperature or smell.

If a takeover will indeed take place, the *driver fitness* for the next level should be *assessed* and *improved* if necessary. For example, the driver can be urged to stop the distracting activity. Again, such communication should be independent of the current driver activity.

Finally, the HMI should *guide the takeover* by supporting the driver in creating *situation awareness* and resuming *vehicle control*. For example, directional lighting can be used to focus the driver's attention on the steering wheel and windscreen, in combination with explanatory HUD or display icons. In addition, communication via LED lights on the steering wheel could be used. Different patterns and colours can communicate for example automation level, upcoming takeovers and nudges to put the hands on the steering wheel. The takeover ritual ends by communicating the *completion* of a successful take over and informing the driver of the *new driving situation*.

A still open research question is how to handle different types of control transfers between levels of automation. I.e., should jumps between levels be allowed, or should takeovers only include from and to manual driving? And if jumps between levels are allowed, should they always go step by step (CM to SB and SB to TtS) or are larger steps also allowed (CM to TtS)? These questions will be answered at a later stage when “in-person” driving simulator experiments are again more feasible due to relaxed COVID-19 regulations.

3.4.4. CM switch on/off

Switching between driving with CM and manual driving can happen instantly, as in this automation level no significant takeover time is guaranteed by the automation. This means that the HMI will not have time to perform an extensive takeover ritual as described in the previous section. Instead, the driver will need to be *informed* on the new *automation status*.
Switching from manual driving to driving with CM can either be initiated by the driver or by the Mediator system, as described in use case 6. The HMI thus needs to *communicate the availability* of this automation level and provide a way to *manually switch on CM*. For a Mediator initiated switch on of CM, the previously described negotiation routine should be used.

3.4.5. Preventive actions

Literature research was performed on corrective actions previously investigated to mitigate driver distraction and fatigue. This literature research provided ideas for the design of both preventive and corrective actions to, respectively, maintain and improve driver fitness. The preventive actions can be divided in four groups: 1) improve understanding of the system, 2) keeping the driver in an alert state, 3) support execution of driver task (monitoring), 4) limiting distractions. Each of these groups affect the driver fitness in different ways.

By improving the understanding of the system, the driver can anticipate potential automation failure and overreliance can be reduced. This in turn can motivate the driver to maintain its fitness at the appropriate level. An example of such a preventive measure is training of driver to use the vehicle and help them understand the automation limitations. In general, an imperative condition for HMI design is that the driver should understand the automation system, fully and intuitively.

Keeping the driver in an alert state helps the driver maintain the resources to perform the task. Examples of preventive measures to remain in an alert state are engaging drivers in conversation or providing other occasional stimuli such as sounds and lights. In addition, adjusting the in-vehicle temperature or smell can improve alertness. Finally, the monitoring task can also be made less monotonous by periodically adjusting the speed, keeping the driver more alert. Generally, however, these preventive measures to keep the driver alert should not be too regular, as drivers can get used to such rhythmic stimuli, causing them to lose their effect. Also, the stimuli should not be unnecessarily intrusive. In higher levels of automation, for example, drivers are allowed to perform secondary tasks and stimuli for preventive actions should not interrupt those tasks. In lower levels of automation, however, stimuli might be somewhat more intrusive as to keep the driver alert and involved with the driving task. Care should still be taken, though, to not create annoyance.
The monitoring task in particular, can also be supported via HMI design, such that task performance is improved and workload reduced. Examples of such HMI designs are highlighting (latent) hazards in the environment via head-up displays to aid drivers in maintaining situation awareness and gamification via head up displays to focus the driver to look on the road.

Finally, distractions can be limited by blocking the execution of secondary tasks so that drivers can focus on their primary driver task. For example, during driving in CM the HMI can completely block certain apps or complete usage of nomadic devices.

In the MEDIATOR project the preventive measures focus on *improving understanding of the system* and *keeping the driver alert*. During CM a stimulus response task will provide *occasional stimuli* to maintain driver alertness. The exact stimulus type is not yet defined, but examples are providing route information or personalised trivia questions or facts. Additionally, the driver's *reaction to the stimulus*, such as reaction time, will be measured and serve as input to the driver state estimation module to improve estimation accuracy.

During driving in SB and TtS, as here the driver is completely out of the driving loop for periods of time, improving *system transparency* and avoiding *mode confusion* and *overreliance* are a main focus. To this end, the HMI will unobtrusively *communicate time budgets* such as minimum takeover time and likely time the current level of automation will be available, as well as information on *reasons for automation fitness to change*. The aim is to create an ambience that reflects the current driver responsibility, which should also be perceived when NDRT's are performed. The communicated information can be used by the driver to anticipate automation changes and better understand the automation limits. For long term planning of NDRT's also information on *route progress* and available automation levels along the route should be communicated. Finally, to improve the driver's understanding of the system, information on *upcoming manoeuvres* and *automation perception*, such as other road users and traffic signs, can be communicated.

3.4.6. Corrective actions

As mentioned, the literature research on preventive actions also provided insight into possible corrective actions to improve driver fitness after (early stage) degraded performance is detected. Three corrective action groups were found: 1) warnings, 2) advises and 3) blocking actions.

Research on warnings shows that they can aid in counteracting distraction and task related fatigue. Sensory modalities to communicate these warnings generally include auditory, visual and vibro-tactile channels. Other counter measures can only be communicated through advises to the driver. Research shows that taking a break and eating, drinking caffeinated beverages or exercising can improve driver alertness after the break. Such advises can be communicated in different ways, such as via display icons or text or via voice messages. Both the warning and advises can be applied in CM and SB driving situations. For both these communication tools to be affective and not decrease driver comfort, however, their *frequency* should not be too high and the *urgency* should be communicated appropriately. E.g., low urgency warnings might require less intrusive modalities than high urgency warnings. If the driver does not respond well to warnings or advices, another corrective action can be to block the usage of distracting advices or force a fatigued driver to rest by dimming lights. However, such more invasive corrective actions can significantly reduce driver acceptance of the system and should be implemented with care.

Next to improving the driver state, the HMI can also aid in direction the driver's attention or actions before a takeover. For example, directional lights or advanced head-up display imaging can direct driver's attention to latent hazards or nudge them to perform certain manoeuvres. However, such measures can also distract a driver, especially in situations where the automation might not have detected all hazards. Additionally, on the long term, it might reduce the driver's capability of gaining situation awareness and decision making due to reduced exercise.

Corrective actions can be initiated at different levels of degraded human performance. Within the MEDIATOR project a focus is on preventing or correcting early stage degraded performance. Additionally, it is expected that degraded performance during automated driving is often related to underload. To this end, the stimulus response task, as described above will also be used for corrective actions. In this case, the *frequency* of providing the stimulus is dependent on the level of degraded performance. I.e., when early stage degraded performance is detected, additional stimulus response tasks will be initiated on top of the regular stimulus response task corresponding to the preventive action.

3.4.7. Driver inputs

The Mediator system might require additional input from the driver for use in different modules. *Interfaces to obtain driver input* should be included in the HMI. For example, basic preference settings for decision logic could be set via the HMI, i.e., a driver can possibly set a preference for switching to automation as often as possible or to only switch to automation if it is available for a minimum amount of time. Another type of input could be to ask the driver for help in improving degraded performance detection, such as confirming a degraded state or stating the hours of sleep that the driver had. Finally, it is possible that wearable fitness devices and other user devices such as phones can aid the driver fitness detection algorithms and the HMI. If the Mediator system is indeed to be connected to those devices, an HMI should be developed to assist the driver with correctly connecting their devices to the Mediator system.

Most of these exact required user input functions are still unknown but should become clear in the next stages of the project and followed up on by the HMI design team.

3.4.8. Integrated HMI

Figure 10 presents an overview of the HMI module. On the left side the inputs (bottom) and outputs (top) to the central component of Figure 2 are shown. The input includes requests for action from the decision logic module, e.g., take over or fitness improvement requests, driver and automation fitness information and other context information relevant for the HMI functions. The HMI module output contains HMI status information that can, for example, be used by the decision logic to monitor action progress, as well as driver input related information such as preferences and settings. The middle of the figure shows the software component, while the right side of the figure shows the hardware components of the HMI. In the software component the input is processed, and the requested action(s) and urgency level(s) are mapped to the appropriate negotiation routine and HMI functions. The execution of the functions is then performed through the hardware component, which consists of HMI controls to communicate with the driver and driver input devices to receive information from the driver.

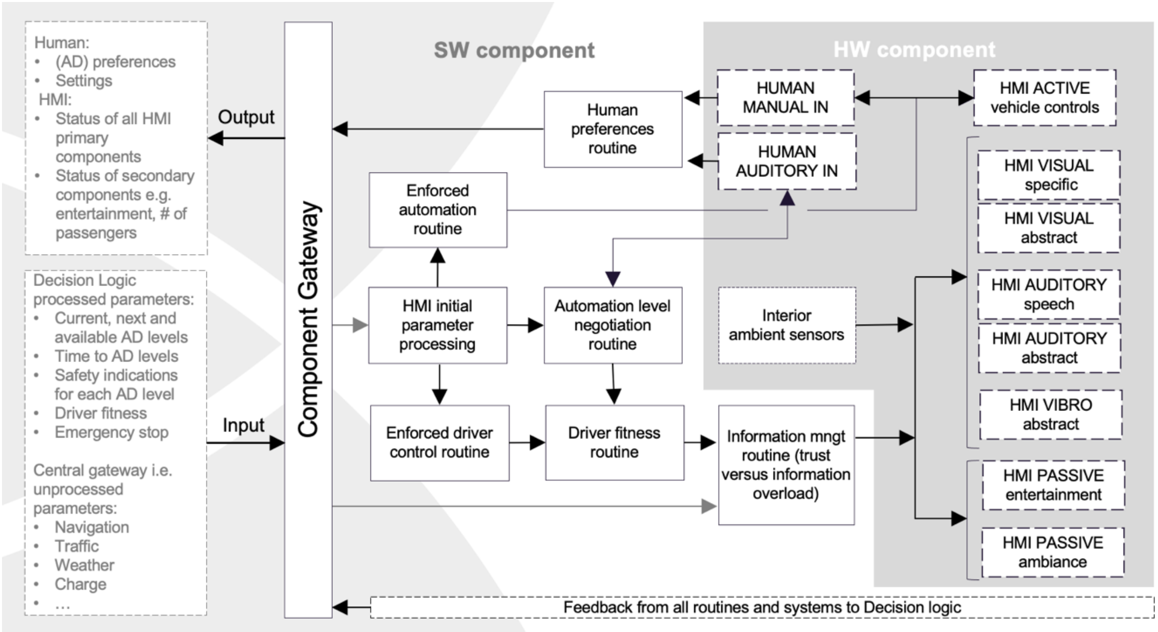


Figure 10: Overview of HMI module

Starting point of HMI design is a conventional automotive interior, as is required for conventional driving. This HMI design will be analysed for necessary and obsolete functionalities. This analysis provides constraints and possibilities for the holistic Mediator HMI.

As user acceptance is one of the main non-functional requirements for the Mediator HMI, personalisation options will be added where possible. The intent of the Mediator HMI is to provide the human with a set of preferences to personalise the vehicle. These will include conventional setting, such seat adjustments, entertainment settings or interior lighting colours. Considering, we are designing a holistic HMI, for each setting and component personalisation must be designed such, that mediator's functionality is not jeopardised.

3.4.9. Key performance indicators

The HMI module functions include conventional driving task HMI functions, guiding takeovers and negotiations, execute corrective and preventive actions and detect and transfer driver inputs. To assess the performance of these functions the following key performance indicators can be used.

- Usability: e.g., System Usability Scale (Brooke, 1986)
- Acceptance: e.g., Van der Laan Acceptance Questionnaire (Van der Laan, 1997)
- Workload: e.g., NASA RTLX (Hart,1998)
- Trust: e.g., Automation Trust Scale (Jian, 2000) and/or measures based on gaze behaviour (Hergeth, 2016)
- User experience: e.g., User Experience Questionnaire (Laugwitz, 2008)
- Mode awareness: e.g., similar to the subjective measures described in Kurpiers (2020)
- Overreliance: e.g., similar to the objective measures described in Kurpiers (2020)
- Takeover quality: e.g., a combination of time to collision, longitudinal and lateral acceleration and time to lane crossing.
- Average fitness increase after corrective action
    - Both distraction and fatigue increase on the same scales as used in the driver module
- Average fitness increase while driving with preventive action as compared to without

- Both distraction and fatigue increase on the same scales as used in the driver module

3.4.10. Functional Requirements

All functional requirements are listed in this paragraph. First a set of general HMI requirements is provided, followed by requirements that are most relevant to only a selection of the use cases. The necessity for implementation of each functional requirement is indicated based on three levels ranging from most to least important: 1) ‘must’ indicates that implementation is mandatory, 2) ‘should’ indicates that implementation is desired, and 3) ‘will’ indicates that implementation is somewhat desired.

General HMI functional requirements:

- The HMI module must perform all *HMI functions of the original vehicle HMI*
- The HMI should unambiguously *make the driver aware of current automation levels*, and therefore the driver’s appropriate sense of responsibility, as consistently as possible through its primary and secondary (ambient) look and feel.
- The HMI must as much as possible fulfil its interaction with the driver, within modes such as preventive and corrective actions, as well as in transfers between modes, through a *single, recognisable and predictable ritual*, for quick and intuitive learning.
- In case the driver indicates a different automation preference than that of DL, the HMI should *negotiate with the driver*. For low necessity levels a seductive negotiation between automation and human is applied, while for higher levels a persuasive negotiation is applied, or even a forced take-over (no negotiation).

Use case 1, forced handover to automation

- In use case 1, when the driver hands over to automation, the system must deliver confirmation feedback.
- In use case 1, during driving in CM or SB, when the driver does not respond to an alert, the system will slow down and eventually stop.
- In use case 1, when the automation wants to take over control and the automation gives recommendations to the human, Mediator should take into account that the human can ignore recommendations although distraction or fatigue is detected.

Use case 2, driver indicates to take back control

- In case the human resumed control (use case 2) while the transfer is executed, the HMI should remain giving feedback regarding mode change and duration.
- In use case 2, when the driver resumed manual control, the driver will be supported in tactical decision making by the HMI.

Use case 3 & 5, comfort and system-initiated takeover

- In use case 3 & 5, if there is disagreement about who is overtaking control, driving manually or in CM, the HMI should inform about the traffic situation in order to proof that there is nothing to gain by overtaking.

Use case 4 & 5, corrective action and human takeover

- In use case 4 & 5, when driving in SB or Tts, the HMI will secure and train unexperienced drivers by asking to take-over in non-critical situations.

Use case 4 & 8, corrective action in SB & CM

- In use case 8, if a driver is detected as distracted for at least 2 seconds at CM, the system should prompt by using warning signals.
- In use case 8, if the system, at CM has already prompted (due to driver distraction) and the driver did not react within 5 seconds, the alert should change be accompanied by *multimodal signals* for another 5 seconds.

- In use case 4 & 8, if the system provides an alert for a distracted driver, then the location of the alert should be in the *driver's field of view* (e.g., tablet or cellular screen).
- In use case 4 & 8, if a driver's visual attention to the driving scene has been detected as inadequate, the system will prompt the driver to perform a task in tandem with driver monitoring.
- In use case 8, if a driver is detected as drowsy/ fatigued at CM, the system should prompt by an audio alert and flashing up an unequivocal instruction on the display.
- In use case 4 & 8, if at CM or at SB the driver is detected as drowsy or fatigued, the system should suggest the driver to take a rest brake and to recommend her to drink coffee and perform light-intermediate physical activity.
- In use case 4 & 8, if a driver at CM or SB does not respond to an alert (detected as unresponsive) when detected as fatigued, the system will slow and eventually stop.

Use case 5 & 9, system initiated human takeover

- In case the human has to take control (use case 5) when the urgency level is high, the takeover request must be by means of intrusive communication stimulating *multiple senses*.
- In use case 5a, while driver engage in NDRT, the system must deliver which mode is currently activated.
- In use case 5a, when driver receive emergency take-over request the system must deliver the importance of immediate driver action is required.
- In use case 5, when in SB a take-over is due, but the human wants to extend AD, the automation must continue outside the available period and should perform an unwanted controlled stop (The type of stop depends on the situation to be out of danger for other traffic).
- In use case 5, when in SB a take-over is due, but the human seems unfit to take-over, the automation is forced to continue outside the available period and should perform an unwanted controlled stop (The type of stop depends on the situation to be out of danger for other traffic).
- In case there is a short time frame for take-over (use case 5&9), the HMI will train the driver (in order to gain experience with take-over situations) to reduce the take-over time.
- In use case 9, when a human, driving in CM, needs to take over, but has too high expectations of automation and therefore becomes distracted with other NDRTs, the system should take into account that the human can still ignore warning signals and/or take over requests.

Use case 7, preventive action

- In use case 7, if a driver at CM handles mobile or other apps, the system will suggest *blocking the apps* that are considered a distraction.
- In use case 7, if at CM, the driver is predicted to become drowsy/fatigued, the system will suggest the driver to engage with a relatively low demanding (secondary) task (audio task such as conversation with Vid) to mitigate fatigue.
- In use case 7, if at SB, the driver is predicted to become drowsy/fatigued, the system will suggest the driver to engage with NDRT (such as game) to mitigate fatigue.
- In use case 7, while driving in CM the HMI must communicate the current mode continuously.
- In use case 7, while driving in CM the HMI should support driver's vigilance through preventive mediation.
- In use case 7, while driving in CM the HMI should make the driver aware of the limitations of the current mode.
- In use case 7, while driving in CM the HMI should employ corrective measures and/or enforce breaks or limit the availability of partial automated driving when needed.

- In use case 7, if a driver has never used partial automation before, the HMI should inform the driver about the limitations of partial automation and about what is required of the driver.

Use case 10, smooth transition TtS to SB

- In case the human has to take control after TtS (use case 10) while awakening the driver to prepare for the transfer, non-intrusive (design) interventions should be used.
- In case the human has to take control after TtS (use case 10) while the SA is regained, the SA must remain, and the human should be guided in order to get prepared for takeover.
- If a TOR is upcoming and the driver is engaging with a non-driving task (use case 10), the system should alert by using *ambient peripheral cues* with an indication for the desired steering direction.
- If a TOR is upcoming (use case 10) the system should provide an early/pre alert.
- If a TOR is upcoming (use case 10) the alerts should be provided with different frequencies based on the immediacy (Time for TOR) of the situation.

Applicable to several use cases:

- In case of a transfer of control (use case 1, 2, 3, 5, 6, 9 or 10), mode confusion will be avoided when the number of possible mode switches is limited by communicating no more than 3 overarching driving modes to the human.
- In case of a transfer of control, from either automation to the driver or from the driver to automation (use cases 1, 2, 3, 5, 6, 9 or 10), when the DL disagrees with the transfer it should communicate this by means of force feedback.
- In all use cases, when driving, Mediator should take into account disagreement of speed (limit) and both human drivers (as Mediator) should be able to give their preference during AD to point out (inconsistent) driving speed during manual driving.
- In all use cases, when using ADS, Mediator should give the human confidence in the ADS and when to access which automation level in order to give guidance in the amount of automation levels related to the suiting traffic situations/environments.

The following functional requirements are applicable while driving in SB or TtS and therefore are also applicable to all use cases which include SB and/or TtS. These functional requirements are detailed below.

- While driving in SB or TtS the HMI must communicate the current mode continuously.
- While driving in SB or TtS the HMI must communicate the time left in current/time to next mode continuously.
- While driving in SB or TtS the HMI should communicate what the next mode will be.
- While driving the option to switch on SB or TtS will only be offered if it is likely that it will be available for at least 4.5 minutes
- When the current mode will change to another mode the HMI should communicate the reason for this change in advance.
- While driving in SB or TtS, the HMI should nudge the driver in what to do.
- While driving in SB or TtS the HMI should communicate the foreseen automation status throughout the route.
- While driving in SB or TtS the HMI should communicate maneuvers that the car will perform in the near future.
- While driving in SB or TtS the HMI will communicate reasons for maneuvers that the car will perform in the near future.
- While driving in SB or TtS the HMI should communicate on automation perception.

- While driving in SB or TtS and if the current mode allows for a setting on presented information to be changed the HMI should provide the option to have its settings on presented information changed.
- If a driver has never used the HMI before the HMI will guide the driver through all its functionalities and how these functionalities relate to the capabilities of the automation.

Finally, a set of non-functional design requirements was also identified.
The system shall:

- The system shall make use of learned, familiar and generally known affordances to *minimize learning effort.*
- The system shall *preserve human autonomy*, as that is a pivotal comfort component and crucial in achieving user acceptance.
- The system shall have *high usability, user acceptance* and *trust* and provide a good user experience.

3.5. Decision Logic

The decision logic is the software module that translates driver and automation state information from the driver and automation modules into Mediator action requests required for a safe and comfortable driving experience. These requests are then sent to either the HMI or the automation module where execution of these actions takes place. To this end, the decision logic module has the following subfunctions:

1. Decide who is fittest to drive based on optimizing safety and comfort
2. Select actions that enable the fittest to perform their (part of the) driving task
3. Monitor action execution
4. Adjust action or action parameters if needed

These functions can be divided in high-level functions (1&2), which generally are dealing with large time steps and low-level functions (3&4), which deal with smaller time steps. To execute these functions the module gets inputs from all other modules and outputs the desired action to the relevant module. In Figure 11, an overview of the inputs, outputs and subfunctions of the decision logic module is shown.

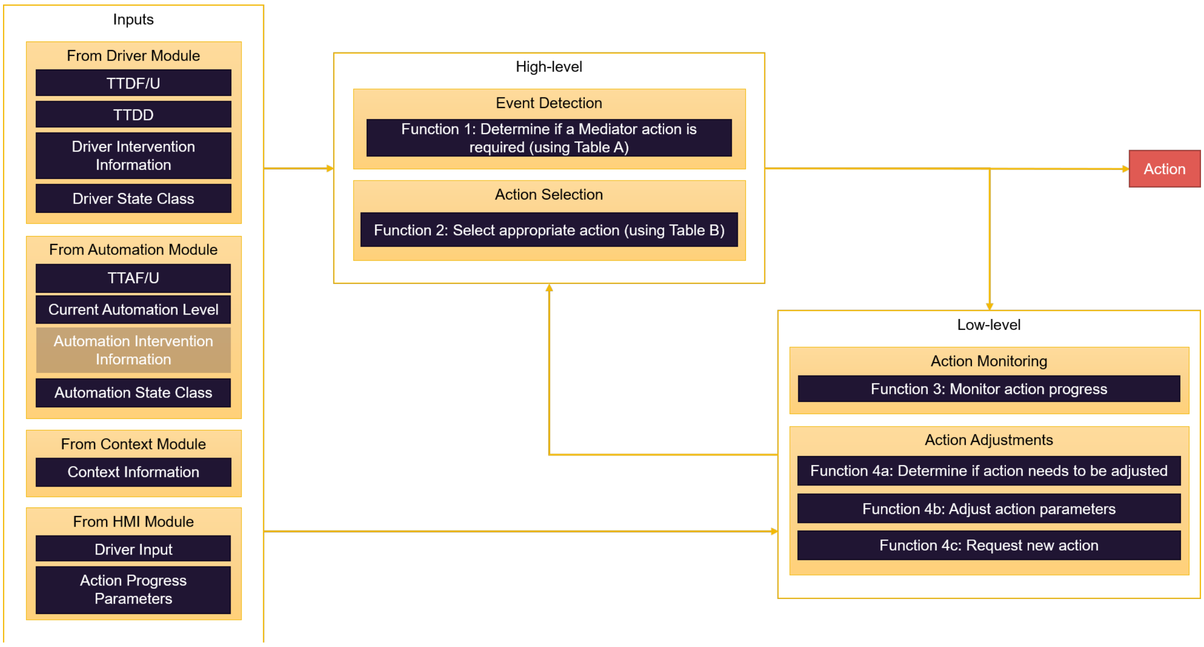


Figure 11: Overview of decision logic module

The decision logic subfunctions are further explained in sections 3.5.1 till 3.5.3. It should be noted that these sections describe the view of the decision logic function requirements and provide initial ideas towards implementation as thought of in this stage of the project. However, this is not set in stone and during the next phase of the project, some functions can be removed or added and, especially implementation related, details are likely changed. To assess the performance of the decision logic a set of key safety indicators is developed and described in section 3.5.5. Finally, the more detailed functional requirements for the decision logic are described in section 3.5.6.

3.5.1. Event detection

The basis of the Mediator system is to determine who is fittest to driver based on the current and future driver fitness, driver comfort and automation fitness. To compare these constructs the variables time to driver (un)fitness (TTDF/U), time to driver (dis)comfort (TTDC/D) and time to automation (un)fitness (TTAF/U) were defined. Using these variables, situations can be identified where driver safety and/or comfort might be at risk and in which a Mediator action is required. When such an event is identified by the decision logic, this same logic will select an appropriate action to mitigate the risk.

For event detection, the current automation level determines which variable or comparison are most relevant to monitor. In general, for manual driving and driving in CM the TTDF should be zero, i.e., the driver should be fit to drive, while for driving in SB or TtS the TTDF should be smaller than TTAU, i.e., a takeover to manual driving should be possible before the automation becomes unfit. In addition, when driving manually or in CM the TTDU should be smaller than TTAF or smaller than the time it takes to successfully execute a fitness improvement action.

However, some note should be added here. When driving in TtS, in theory the automation is expected to be the fall back and thus be able to bring the vehicle to a safe place to park. In this case, the driver is therefore not required to serve as back up, but not doing so can result in driver discomfort due to interruption of the trip. When driving in TtS, an upcoming end of the automation ODD should therefore at least be considered as expected to result in driver discomfort. Another option is to adjust the definition of TTAU in this case to the time until the automation will initiate its fallback option.

Apart from initiating actions due to safety relevant events, decision logic should also consider initiating an appropriate action when driver discomfort is expected in the near future. Within the scope of the MEDIATOR project, when driver discomfort is expected, either switching to manual driving or to automated driving is considered as an appropriate action to mitigate driver discomfort and no distinction is made between the levels of automation. The suggestion to transfer control should occur at a comfortable time before driver discomfort is expected.

Finally, if a user request occurs, the decision logic should always respond with an appropriate action and if the automation level changes specific actions might also be necessary.
A first effort to describe the conditions in which no Mediator action is required is summarised in Table 10. The decision logic should *monitor the conditions* and if a violation occurs, the decision logic should assess if an appropriate action can be taken and if so, which action is most appropriate.

Table 10: Event detection conditions

Automation Level	TTDF	TTDU	TTDD	User request occurred
Manual	TTDF = 0	TTDU < time to improve fitness *or* TTDU< TTAF	TTDD> comfortable takeover time	No
CM	TTDF = 0	TTDU< time to improve fitness *or* TTDU< TTAF	TTDD> comfortable takeover time	No
SB	TTDF < TTAU	-	TTDD> TTDF	No
TtS	TTDF < TTAU	-	TTDD> TTDF	No

The TTD variables used for event detection are all estimated with a *confidence interval* defined by the worst, likely and best case. The decision logic should take this interval into account during the event detection and especially, the timing of the action.

If the confidence interval of the driver fitness related parameters is too high to make a decision, one idea is to have the option to request additional input to the driver with the aim of reducing this confidence interval. The way in which this is now foreseen is to initiate the stimulus response task used for corrective and preventive actions, as described in sections 3.4.6 and 3.4.5, and use the driver responses to increase the reliability of the fatigue and distraction estimates. Another option, not currently foreseen, could be to simply ask the driver for a rating of their fatigue or distraction level. For the decision logic, however, it is mainly important that the algorithm structure allows for such *requests to decrease the confidence level*.

To summarise, the action detection aspect of the algorithm thus needs to monitor the conditions under which no Mediator action is required while taking into account confidence intervals of the relevant parameters and requesting input to decrease the confidence intervals if necessary.

3.5.2. Action selection

The appropriate actions that the decision logic should select depend on the event that was detected and the current automation level. As mentioned in section 2.3.1, the events can be grouped as follows: driver degraded performance, automation degraded performance, driver expected discomfort, driver requests and new automation level. Each of these events can occur while driving with different levels of automation. In Table 11 an overview is given of these events, the automation levels they can occur in and the high-level actions that are appropriate to take to maintain a safe driving situation. The safety-appropriate actions are divided in those described by the use cases and other appropriate actions. The other appropriate actions are generally similar to the actions described in the use cases corresponding to other automation levels. However, the action “improve automation fitness”, as mentioned in section 3.2.1.3, is not part of the MEDIATOR project scope. The same holds for the action “adjust automation settings”. While currently out of scope of the MEDIATOR project, the literature research on uncomfortable driving situations described in D1.2 revealed that adapting the automation driving style can in certain cases improve driver comfort. Taking these possible actions into account when setting up the decision logic structure can aid future developments of the Mediator system.

Table 11: Decision logic actions for detected events and automation levels.

Event	**While driving**	**Decision Logic Safety Appropriate Actions**	
		Action from use case	**Other**
Driver degraded performance	Manual	Takeover to SB (UC 1)	Improve driver fitness, takeover to TtS
	CM	Improve driver fitness (UC 8)	Takeover to SB/TtS
	SB	Improve driver fitness (UC 4)	Takeover to TtS
Automation degraded performance	CM	Switch off CM (UC 9)	Improve automation fitness
	SB	Takeover to manual (UC 5)	Takeover to CM, improve automation fitness
	TtS	Switch to SB (UC 10)	Takeover to manual/CM, improve automation fitness
Driver expected discomfort	Manual	Takeover suggestion to SB (UC 3) or CM (UC 6)	Takeover suggestion to CM/TtS
	CM	-	Takeover suggestion to TtS/SB/Manual *(or adjust automation settings)*
	SB	Takeover suggestion to manual (UC 2)	Takeover suggestion to TtS/CM *(or adjust automation settings)*
	TtS	-	Takeover suggestion to SB/CM/Manual *(or adjust automation settings)*
Driver request		Check request safety and execute if safe	-
New automation level	CM	Initiate stimulus response task (UC 7)	-
	All	Change in-vehicle ambience/HMI configuration according to automation level	-

Table 11 shows that for each combination of event and current automation state, several possible actions can be appropriate. To decide which action is most appropriate for each situation, more information than just the event and current automation level is needed.

First of all, the *probability of success* for each action should be assessed. For example, a takeover to automation as a solution to driver degraded performance will not be successful if this automation level is not available before the driver becomes unfit. A takeover to automation for comfort reasons, on the other hand, might have a different probability of successfully increasing comfort for different situations and drivers. Not only the probability of a successful outcome, but also the *duration* until the outcome is reached has uncertainties. This is especially true when driver fitness improvement is involved. For example, the probability of successfully improving driver task related fatigue up to the desired level might be higher after 1 min than after 10 seconds. For this reason, the probability of success for such actions is described with a worst, likely and best-case scenario, where the best-case scenario describes the shortest time within which the driver state could be improved. Apart from safety related considerations, also the driver comfort should be taken into account. For example, if a certain action results in only a minimal acceptable safety reduction, but increases driver comfort significantly, this action could have the preference over an action that does not change safety but reduces comfort significantly. The action selection should thus be *optimised for driver safety and driver comfort*, while *constraining for safety* to avoid truly unsafe situations. For both safety and comfort reasons, the optimisation should not only take into account the current situation, but also *past situations* and possible *future situations*. For example, if a suggestion to take over was just rejected by the driver, this same suggestion should not be initiated again unless the situation significantly changed. Or regarding future situations, if in the near future a takeover to manual driving will happen, it is unnecessary to initiate a separate request to improve driver fitness due to early signs of degraded performance. In this case, it might be more comfortable to simply start improving driver fitness somewhat later as part of the takeover routine. Overall, the *frequency of interactions* during a trip should also be kept to a minimal. For example, the decision logic should not initiate a takeover to SB when it expects that a takeover back to manual driving will happen shortly after.

To summarise, the objective function and constraints description of the optimisation algorithm need to take into account both safety and comfort of the driver, the probability of success of a certain action, past and future situations and the frequency of interactions while constraining for safety. Each of these aspects will need to be weighted according to their importance. While some indications to the relative importance can be estimated at forehand, the exact parameters of such algorithm should be estimated by comparing desired and actual outcomes of the algorithm for different situations.

3.5.3. Action monitoring and adjustment

When an appropriate action is selected, this is generally sent to the HMI module to be executed. Each of these actions can contain several sub actions that are executed by the HMI. For example, the takeover procedure might start with informing the driver, then improving his or her fitness and only then a request to turn off automation is sent. Each of these steps should be monitored to determine if the progress is as expected by the decision logic. If this is not the case, or if other factors change the situation, the decision logic should be able to either *adjust the action parameters* and/or the *action* itself. To monitor actions that include driver fitness improvement the expected progress over time, described in Table 5 from the driver module, can be used. For the control transfer to automation, other checks, such as a confirmation from the driver via the HMI could be used. The exact implementation of this monitoring and adjustment process is not yet defined and parts of it can happen either in the HMI module or in the decision logic module.

3.5.4. Personalisation

One of the focus points of the MEDIATOR project, as described in section 2.2, is personalisation of the Mediator system. The decision logic module needs to support this in several ways. First of all, as mentioned in section 3.1.5, an adjustment of the probabilities of improving comfort in certain situations, as described in Table 4, can be done online for each driver using information on their rejection and acceptance of takeover suggestions. This means that the decision logic should be able to *obtain acceptance/rejection information from the HMI* and to *adjust relevant parameters in* *the comfort table*. One idea for adjusting the probabilities based on user rejection or acceptance is to use a logarithmic scale, where initial rejection/acceptance cause large changes, but later ones result in only small changes. In this way the comfort increase/decrease will also never reach the maxima.

A second way of personalizing the decision logic is to *adjust the weights in the objective function* based on acceptance and rejection of takeovers. For example, for a driver who often rejects a takeover to automation as a result of degraded driver fitness, but instead rapidly improves their fitness, the relative weights for the actions “improve driver fitness” and “takeover to automation” could be adjusted.

Finally, certain algorithm parameters, such as the objective function weights, could be pre-set according to a certain driver profile that can be chosen by the driver him or herself. A very simplified example of a decision logic algorithm that uses such approach can be found in one of the master theses related to the MEDIATOR project (Vermunt, 2020). In this case, the decision logic should be able to *read out a table of driver specific parameters* that was selected by the user via the HMI.

It is currently still undecided which of these personalisation options will surely be implemented in the decision logic.

3.5.5. Key Performance Indicators

Validating the decision logic, and the Mediator system in general, will prove a challenging task, as no clear baseline exists, and the scope of all possible scenarios covered by the use cases is still rather large. Especially for the decision logic and HMI modules, the use cases describe mainly one outcome of the scenario, e.g., the driver fitness improvement action is initiated and this is successful, but the modules should take into account deviations from these scenarios as well, e.g., the fitness improvement does not go as planned.
However, a first step in the validation process is to define the key performance indicators according to which the system will be evaluated. For the decision logic a first set of KPI’s was identified. The KPI’s refer to performance measures of the decision logic algorithm when running it for the 10 use cases with dummy inputs from all other modules. An initial list of proposed KPI’s is as follows:
Proposed KPI’s:

- **Number of safety critical events**
    - A safety critical event can be described by an area on the driver/automation fitness plane as explained below and is hereby proposed to be split in three criticality levels.
    - *High performance*: low number of events
- **Total time in safety critical situation**
    - This includes the total time during the simulation that the combination of driver and automation fitness measures were classified within one of the levels of safety criticality.
    - *High performance*: low time spend
- **Average duration of safety critical events**
    - The average time per safety critical event spend in a state where the combination of driver and automation fitness measures were classified within one of the levels of safety criticality.
    - *High performance*: low duration
- **Number of times no solution was found**

  - The number of times that the decision logic algorithm did not converge to a viable solution throughout the simulation. Hereby also the reason for this non-convergence is of importance, i.e., if this happened because a safety criticality level 3 situation occurred this should be dealt with differently than when it happened due to any other reason.
  - *High performance*: low number of times (ideally zero)
- **Number of events where comfort was improved**
  - The number of times the decision logic selected an action to improve driver comfort.
  - *High performance*: high number of events
- **Total time in uncomfortable situation**
  - The total time spend in a situation of which the Mediator system expects the driver to feel discomfortable, based on the uncomfortable situations described in Table 4.
  - *High performance*: low total time
- **Average duration of uncomfortable event**
  - The average time per event the driver spends in an uncomfortable state.
  - *High performance*: low duration
- **Number of actions**
  - Total number of actions initiated by the decision logic. It is probably informative to subdivide this into the action types.
  - *High performance*: low number of actions
- **Total time without actions**
  - The total time during the complete simulation where no action was being executed.
  - *High performance*: long total time
- **Average time between actions**
  - The average time between the end of a first action and the start of a second action.
  - *High performance*: long average time

A **safety critical event** can be described using the fitness plan as shown in Figure 12. Here it is proposed to use three levels of criticality, but this can be adjusted at a later stage. A safety critical event is said to occur when the combination of driver and automation fitness is projected to a point within one of the areas with a safety criticality level of 1 or higher. Level 1 criticality applies when both driver and automation have degraded performance but are also both still able to perform the driving task. Level 2 criticality occurs when either driver or automation is not fit the driver and the other has degraded performance. Finally, level 3 criticality applies when both driver and automation are not able to perform the driving task.

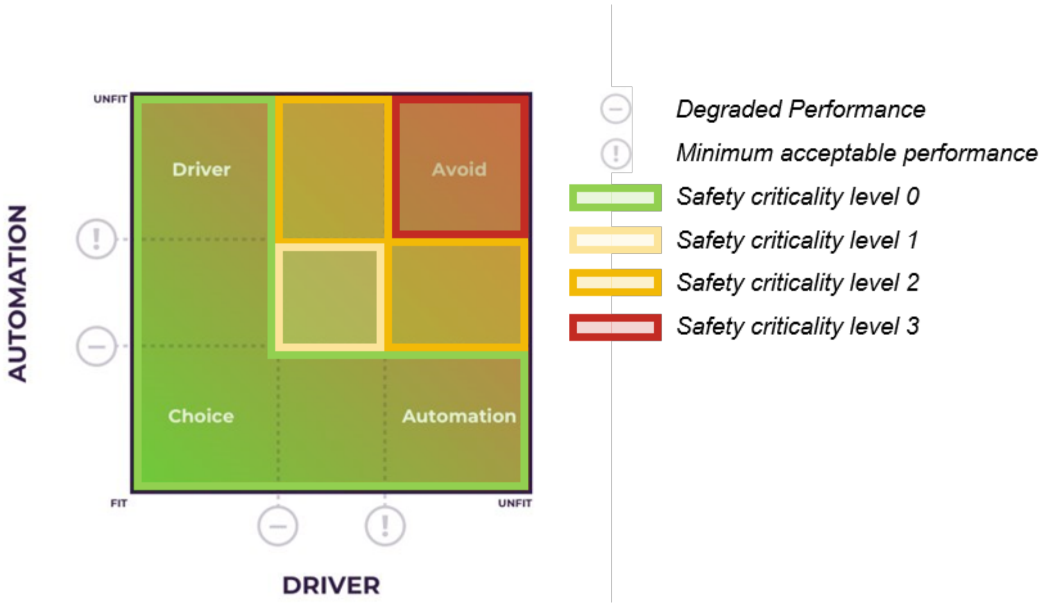


Figure 12: Driver/Automation fitness plane with safety critical events

3.5.6. Functional Requirements

The previous sections describe several functional requirements of the decision logic. Here an overview of these requirements is provided, starting from the main functions and subdivided in more detailed descriptions of the corresponding subfunctions.

The decision logic should:

- Detect situations where a Mediator system action is required
  - Monitor the conditions described in Table 10.
  - Assess if conditions are met taking into account uncertainties
  - Request input via the HMI to decrease such uncertainties
- Select the appropriate action
  - Estimate the probability of success of the actions described in Table 11
  - Optimise action selection for driver safety and comfort, which comes down to optimizing the proposed list of KPI's in section 3.5.5
  - Constrain the optimisation for safety
  - Personalise the action selection to each driver (ideas on how to implement this were given in section 3.5.4)
- Monitor the action execution
  - Provide other modules with the necessary information to execute the action
  - Collect information on action execution progress from other modules
  - Detect relevant differences between expected and actual action progress
- Adjust the action or its parameters
  - Identify adjustments to the current action parameters that could improve the situation
  - Identify other actions that could be initiated that could improve the situation
  - Communicate adjustments to the current action to other modules

Ethics

Apart from the functional requirements that result from the Mediator system function descriptions, also non-functional requirements, which describe system attributes such as reliability, should be considered. Such requirements will possibly translate into more specific functional requirements at a later stage. One source of such non-functional requirements is engineering ethics.

Autonomous Vehicle (AV) should be designed to optimise their actions to reach the best outcome for any given scenario. In reality, AVs have to operate in situations for which a perfect solution could not be identifiable or for which it was not specifically tested, by evaluating scenarios and explore options within contexts and by responding in a way that keep humans as safe as possible. This is why several initiatives are taken around the world to set ethical guidelines for such systems.

The European Commission has established the European Alliance for Artificial Intelligence (futurium.ec.europa.eu/en/european-ai-alliance), with the task of proposing ethical guidelines on artificial intelligence, and the aim of providing a horizontal approach on ethical issues concerning autonomous systems, which also interests automated vehicles. UNESCO has embarked on a two-year process to elaborate the first global standard-setting instrument on the ethics of artificial intelligence (AI), in the form of a Recommendation, emphasising that currently no global instrument covers all the fields that guide the development and application of AI in a human-centred approach (UNESCO, 2020). Finally, the German government has established an ethics commission to establish guidelines for self-driving cars in particular.

While many examples related to such ethics guidelines refer to (vehicle) automation specific decisions, such as vehicle control related dilemmas on where to guide the vehicle in critical situations, some are also relevant for Mediator-like systems, where the main focus is on deciding who should drive. In the following two sections, respectively ethical guidelines related to artificial intelligence and those related specifically to autonomous vehicles are described and their relevance for the Mediator system is discussed.

4.1. Artificial intelligence guidelines

While the UNESCO work refers to general artificial intelligence systems, it can often also be applied to the Mediator system, where there is a multimodal sensing of the environment (by cameras, radar, GPS, laser, etc.) and an artificial system that takes decisions based on independent learning algorithms.

In particular, it arises that the *validity* of an AI-driven decision should be treated with caution because it could be not necessarily fair, accurate or appropriate, as it is susceptible to inaccuracies, discriminatory outcomes, embedded or inserted bias and limitations of the learning process. When validating the Mediator system, inaccuracies in all parts of the system should be considered and appropriate design strategies should be applied. For example, if the driver state is detected as degraded, the decision logic might initiate a takeover. However, if this detection was inaccurate, this might lead to uncomfortable and even unsafe situations. Such issues can be mitigated by on the one hand increasing detection accuracy through fusion of different sources of information (physiological measures, video data and possibly even human input), and on the other hand by taking into account the detection accuracy when making decisions and executing them, e.g., with low detection accuracy only suggest instead of urging the driver to hand over control.

Furthermore, even keeping the human "in the loop" to moderate the machine decisions, may not be sufficient to produce the best choice if *system transparency* is not adequate. In fact, as cognitive AI does not make decisions in the same way as humans would, the driver could not be prepared with the proper amount of information (s)he would need in order to decide if the data-driven action fulfils his or her intentions. Another key issue for AV is therefore the nature and interpretation of actions proposed by algorithms, which are not always intelligible to humans. Even if the driver knows what information is used by the algorithm that is driving the car, (s)he might not know how the algorithm decided to process the information. This could happen because algorithms can use a multitude of criteria to reach decisions, with the human's consequent inability to know why a particular choice has been made by the system, and this means that the choice might be less likely to be trusted. This issue becomes even more problematic if one considers that the type of algorithms that will be used in autonomous cars are neural networks, that are seen as the most efficient and accurate, but they are also a black box. Furthermore, auto-adaptive and self-learning algorithms, which properly are considered as the most technically advanced solutions, are also not completely predictable, not even by their own developers. (Sütfeld et al., 2017*)*

The debate among experts has led to some first results in terms of defining principles for algorithmic transparency and accountability, with statements and drafts for public discussion about "Ethically aligned design". These results underlined the need for *accountability* that can help in "proving why a system operates in certain ways to avoid confusion or fear within the general public" (IEEE, 2016 – USACM, 2017). For the Mediator system such transparency and accountability considerations should also be taken into account during the validation of the system.

4.2. Autonomous vehicle guidelines

In 2017 the first law in Europe that allows the testing of autonomous vehicles also on public roads was approved in Germany. The law establishes some fundamental principles that could inspire ethical values for the Mediator system:

- The driver must be able to get the control of the vehicle at any time and deactivate the electronic systems.
- In the event of a crash, the ultimate responsibility will lie with the driver and not with the manufacturer.

These principles could be applied in, for example, the HMI design, which would allow the driver to switch off the Mediator system at all times and would make drivers aware of their ultimate responsibility for driving safety. However, it should be taken into account that these principles mainly refer to current technology and might change when SAE level 4 type vehicles enter the market.

The Ethics Commission of the German Federal Ministry of Transport and Digital Infrastructure also provided comprehensive guidelines for ethical decision making for self-driving cars (Ethics Commission, 2017). Some of these guidelines are particularly relevant for the Mediator system and are briefly discussed here. Regarding the trade-offs included in the decision logic module for selecting certain actions over others, the guidelines are clear that safety should come first:

"The primary purpose of partly and fully automated transport systems is to improve safety for all road users" (from Guideline 1)

and

“The licensing of automated systems is not justifiable unless it promises to produce at least a diminution in harm compared with human driving, in other words a positive balance of risks.” (from Guideline 2)

The guidelines also support MEDIATOR project focus on *preventing* safety critical situations:

“Based on the state of the art, the technology must be designed in such a way that critical situations do not arise in the first place.” (from Guideline 5)

Several guidelines refer to the HMI design of automated vehicles, showing the importance of this aspect of vehicle automation. The focus on human-centred design is also advised within the guidelines:

“To enable efficient, reliable and secure human-machine communication and prevent overload, the systems must adapt more to human communicative behaviour rather than requiring humans to enhance their adaptive capabilities.” (from Guideline 17)

Another focus of the MEDIATOR project, namely avoiding mode confusion, is also mentioned in the guidelines. Additionally, they mention that documentation of who is responsible when is important:

“In the case of non-driverless systems, the human-machine interface must be designed such that at any time it is clearly regulated and apparent on which side the individual responsibility lie, especially the responsibility for control. The distribution of responsibilities (and thus of accountability), for instance with regard to the time and access arrangements, should be documented and stored. This applies especially to the human-to-technology handover procedures. “ (from Guideline 16)

Another HMI related guideline is closely linked to the time budget concepts used within the MEDIATOR project. The guidelines state that abrupt takeovers should be virtually obviated, and time budgets for takeovers should just generally be available:

“The software and technology in highly automated vehicles must be designed such that the need for an abrupt handover of control to the driver (“emergency”) is virtually obviated.” (from Guideline 17)

A still open discussion point within the MEDIATOR project has been on whether the Mediator system should be able to limit the driver to take back control if severe degraded human fitness is detected. While not directly mentioned in one of the twenty guidelines, Chapter IV, section 3 provides some useful thoughts on this:

“One manifestation of the autonomy of human beings is that they can also take decisions that are objectively unreasonable, such as a more aggressive driving style or exceeding the advisory speed limit. In this context, it would be incompatible with the concept of the politically mature citizen if the state wanted to create inescapable precepts governing large spheres of life, supposedly in the best interests of the citizen, and nip deviant behaviour in the bud by means of social engineering. Despite their indisputable well-meaning purposes, such states of safety, framed in absolute terms, can undermine the foundation of a society based on humanistic and liberal principles. (…) Decisions regarding safety risks and restrictions on freedom must be taken in a process of weighing-up based on democracy and fundamental rights. There is no ethical rule that always places safety before freedom.” (from Chapter IV, Section 3)

This section places the emphasis on the important value of freedom that humans have, which might outweigh the safety impact in the situation of driving with severely degraded driver fitness. Finally, as the Mediator system expects to measure large amounts of personal data such as driver face and wide-angle videos and physiological measures, privacy considerations are also important.

The guidelines propose that drivers are always the ones who decide if their data can be used and forwarded.

"It is the vehicle keepers and vehicle users who decide whether their vehicle data that are generated are to be forwarded and used." (from Guideline 15)

5. Validation

Another source for (non-)functional requirements, in addition to the ethics discussed in the previous chapter, is related to the evaluation and validation of the Mediator system. One major challenge is to find appropriate baselines, as no system similar to the Mediator system exists as of yet. Choices for baselines and possible functional requirements that result from these choices are already important during the upcoming development and implementation phase of the Mediator system. The Mediator system validation studies that follow after this phase will make use of several evaluation platforms such as (driving) simulators and Wizard of Oz and real vehicle prototypes. The capabilities and limitations of these evaluation platforms should also be taken into account during the implementation phase. In this chapter those aspects that could be relevant for defining functional requirements of the Mediator system and its validation during the project are discussed.

5.1. Baselines

First of all, when validating a system, generally the designed system is compared to a baseline version. For the Mediator system, however, this proves to be difficult, as no similar systems exist for the automotive application. An alternative option is to focus the validation somewhat more on the subsystems for which baselines might be easier to define.

In both industry and research many different types of driver monitoring systems exist. However, many of these systems can only be applied to manual driving scenarios, as they rely on control behaviour for their assessment of driver fitness. A baseline system for the driver module should instead be applicable in automated driving as well. A comprehensive review of available systems on the market that focus on detection of fatigue and/or distraction can be found in Hermens (2020). Four systems were identified that detect both fatigue and distraction and could thus possibly be used as baselines for the driver module validation. All systems are camera-based systems and use eye and face videos to assess fatigue and distraction.

- The seeing machines system Guardian (www.seeingmachines.com) measures of drowsiness level, distraction events, microsleeps and level of engagement using eye and face monitoring. The level of engagement can be used to determine if a handover from the automation to the driver can be done safely.
- The driver behaviour alerts system from Nauto (www.nauto.com) claims to detect fatigue and distraction based on their camera data. Distraction is detected in three severity levels. Severity levels “mild”, “medium” and “severe” refer to 2.5, 4 and 5.5 seconds of continuous distraction, respectively. They define common distracted behaviour types as Cell Phone & Tablet, Drowsiness, Eating, Paperwork, Stereo, Texting.
- The Eyesight system (www.eyesight-tech.com) detects the states distracted, drowsy or asleep by analysing eyelid, gaze, head position and orientation, blink rate and duration and pupil dilation.
- The driver status system from Streamax (www.en.streamax.com) can detect the driver states fatigued, distracted, phone use and smoking.

The Guardian system from Seeing Machines is the only one measuring levels of drowsiness rather than a drowsy or not drowsy state. The Nauto system is instead the only system measuring different levels of distracted driving. As time to driver (un)fitness, as estimated in the Mediator system, is a continuous variable, these systems might be most applicable as baselines for the driver module.

For the HMI module, baseline systems for subfunctions, rather than for the complete HMI, could be considered as no HMI design exists that covers all use cases. The complete HMI could instead be assessed on user acceptance in general terms, without comparing it to a baseline. Baseline HMIs for subsystems could, for example, be based on HMIs from driver monitoring systems that also provide suggestions for improving driver fitness, such as showing a coffee icon or suggesting finding the nearest petrol station to rest. These HMIs could be used as a baseline for the corrective actions function of the Mediator HMI. For takeover procedures possibly existing takeover procedures from L2 vehicles could be used or HMI takeover designs from other research projects. For preventive actions, a baseline could simply be no preventive action at all.

For the decision logic module, the baseline is difficult to test separately using human in the loop experiments, as such experiments would generally require the HMI as well. Instead, the decision logic module could be compared against a simpler version of the decision logic algorithm, e.g., based on simple “if-else-then” rules. Such simplistic baseline could be used to assess if adding more complexity to the algorithm was indeed necessary.

For the automation module, a baseline is probably not necessary. Instead, the assessment can focus on sufficing criteria. Such criteria would focus on whether the module output suffices, rather than if it is more optimal than other solutions. The main focus is then on whether the module can indeed output the required information and has the corresponding accuracy needed by the Mediator system.

The focus on validation of subsystems rather than on the complete Mediator system brings along the need for *dummy variables*. When designing the different modules and subfunctions, in parallel, realistic dummy variables should be designed for use in the validation on other modules and sub functions that depend on this output. The Mediator system as a whole should in turn be set up in such a way that these modules and subfunctions can be *tested separately* without having to implement hardware or software that is not needed for that particular validation.

5.2. Key Performance Indicators

The evaluation and validation of the Mediator system and/or submodules and functions require assessment criteria in the form of key performance indicators (KPIs). The Mediator system versions that will be used for evaluations will need to *log* all inputs and outputs relevant to determine such KPIs. As some KPI’s might require data from the vehicle system, such as CAN data, another functional requirement is that the Mediator system outputs can be *synchronised* with other data sources. Throughout Chapter 3 KPIs were proposed for each module. An overview of all identified KPIs required to assess the Mediator system are shown in Table 12.

Table 12: Overview of Mediator system KPIs

KPI	Definition
Driver Module	
Driver state detection performance	True/False positive/negatives of the detected driver state and derived measures from them, such as AUC, sensitivity and accuracy • ROC curves can be used to visualise the some of these results • Ground truth measures are difficult to obtain. Subjective measures, such as KSS scores and questionnaire results, will most likely be used. • Thresholds for fitness/unfitness can be based on literature and/or assessed via driving performance measures such as time to collision and lane deviation.
TTDU/F estimation performance	Root mean square error (or similar measure) of time to driver fitness/unfitness/discomfort.
Driver fitness improvement estimation performance	Root mean square error (or similar measure) of the achieved fitness improvement.
Automation Module	
TTAU/F estimation performance	e.g., classification performance measure based on false/true positives/negatives in predicted changes in automation performance and root mean square error (or similar measure) of time to automation fitness/unfitness.
Automation state class	e.g., classification performance measure based on false/true positives/negatives and number of times a more detailed class could be identified than “end of ODD”.
Context information performance	Number and accuracy of context relevant variables required by the Mediator system that can be provided by the automation module.
HMI Module	
Usability	e.g., System Usability Scale (Brooke, 1986)
Acceptance	e.g., Van der Laan Acceptance Questionnaire (Van der Laan, 1997)
Workload	e.g., NASA RTLX (Hart,1998)
Trust	e.g., Automation Trust Scale (Jian, 2000) and/or measures based on gaze behaviour (Hergeth, 2016)
User Experience	e.g., User Experience Questionnaire (Laugwitz, 2008)
Mode awareness	e.g., similar to the subjective measures described in Kurpiers (2020)
Overreliance	e.g., similar to the objective measures described in Kurpiers (2020)
Take-over quality	e.g., a combination of time to collision, longitudinal and lateral acceleration and time to lane crossing.
Corrective action effectiveness	The average increase in fitness after corrective action • For both distraction and fatigue on the same scales as used in the driver model
Preventive action effectiveness	The average increase in fitness while driving with preventive action as compared to without • For both distraction and fatigue on the same scales as used in the driver model
Decision Logic Module	
Number of safety critical events	A safety critical event can be described by an area on the driver/automation fitness plane preferable split in three criticality levels (low, medium, high).
Total time in safety critical situation	The total time that the combination of driver and automation fitness measures were classified within one of the levels of safety criticality.

Average duration of safety critical events	The average time per safety critical event spent in a state where the combination of driver and automation fitness measures were classified within one of the levels of safety criticality.
Number of times no solution was found	The number of times that the decision logic algorithm did not converge to a viable solution. Also, the reason for this non-convergence should be analysed.
Number of events where comfort was improved	The number of times the decision logic selected an action to improve driver comfort.
Total time in uncomfortable situation	The total time spent in a situation of which the Mediator system expects the driver to feel uncomfortable, based on the uncomfortable situations described in Table 4.
Average duration of uncomfortable event	The average time per event the driver spent in an uncomfortable state.
Number of actions	Total number of actions initiated by the decision logic. It is probably informative to subdivide this into the action types.
Total time without actions	The total time where no action has been executed.
Average time between actions	The average time between the end of a first action and the start of a second action.

5.3. Validation outlook

The evaluation platforms, i.e., the (driving) simulators and vehicle prototypes, used for the validation of the Mediator system in WP3, might also impose limitations for the implementation of the system within the MEDIATOR project. Due to their flexibility, simulator platforms likely do not pose many limiting requirements on the Mediator system. Vehicle prototypes used for on road studies, however, need to adhere to many regulations and likely impose constraints on the Mediator system that will be implemented during these studies. Currently, two prototype vehicles are foreseen for this purpose. The first prototype will have automation functionalities but can only be driven by professional drivers. The second prototype will be equipped with a Wizard of Oz system, where the automation is faked by a professional driver trained for this purpose. In such a set up the Mediator system can be tested on naïve participants.

This set up has been used by many automotive companies before, such as FCA, Volvo, Citroen, Volkswagen, BMW and Bosch. Any Mediator system version that is implemented in such a set-up should be able to function appropriately for the chosen use case using human rather than automation input. For example, detection of other traffic will likely not be done automatically, but instead high traffic density could be observed by the experimenter and used as input to the Mediator system. This prototype set-up thus imposes a functional requirement on the Mediator system version used in such prototype in that it should be able to *receive input from the experimenter*.

To test the Mediator system in on road real world scenarios other limitations might apply. For example, each deviation from an approved vehicle for testing on road should be assessed and approved again. When deciding on the exact functions and corresponding required hardware and software to implement in the prototypes for on road testing, the *feasibility* of both implementing the Mediator system in these vehicles and getting such system approved should be taken into account.

6. Mediator Functional Requirements

The previous chapters have discussed the scope of the MEDIATOR project and the Mediator system functions and corresponding functional requirements. This chapter provides an overview of both the functional and non-functional requirements for the Mediator system that were described and thus are identified in this stage of the project. Table 13,

Table 14 and

Table 15 summarise these functional and non-functional requirements. The functional requirements define the function of the system and its components whereas the non-functional requirements define system attributes such as reliability. Both types of requirements provide guidance to the design and development of the technical Mediator components and overall system. Final development, however, will be subject to the constraints given by development time, budget, and available prototype platforms, such that some requirements may not be met completely.

The functional requirements are divided into those that relate to Mediator software functions (Table 13) and those that relate to Mediator system hardware functions (

Table 14). The software related requirements are subdivided into requirements regarding the driver, automation and context state detections, the Mediator decisions, the Mediator actions and the ethics & validation of the complete system. The hardware related requirements are subdivided in those requirements related to sensors or system inputs and those related to controls or system outputs.

The presented requirements provide a basis for the next stages of the project, the development and evaluation of the mediator system. They specify the system to be build and developed (WP2) and the criteria for evaluation (WP3). An updated version of the requirements will be prepared at the end of the project. This updated version will include the new insights gained during the project and could serve as a bases for further development and exploitation of the Mediator system.

Table 13: Overview of software related Mediator system functional requirements

Functional Requirements (software related)
Driver State

The system shall estimate worst, likely and best-case *time to driver (un)fitness* based on driver fatigue and distraction estimates for the current driving context
• The system shall *estimate* driver *fatigue* and *distraction*
    o The system shall detect heart rate, respiratory and facial features
    o The system shall estimate KSS score
    o The system shall detect eyes on/off road
    o The system shall estimate distraction severity based on eyes off/on road
    o The system shall detect non-related driving task
• The system shall *predict* driver *fatigue* and *distraction*
    o The system shall predict fatigue progression based on current KSS score
    o The system shall predict loss of situation awareness based on distraction severity

- The system shall predict time to driver fitness based on non-related driving task involvement

- The system shall estimate when the driver is *unfit to drive*
    - The driver is deemed unfit to drive if it can no longer execute the manual driving task in a sufficiently safe manner due to degraded cognitive abilities (fatigue) or loss of situation awareness (distraction).
- The system shall estimate *time to driver fitness* as the longest estimate based on distraction or fatigue
- The system shall estimate *time to driver unfitness* as the shortest estimate based on distraction or fatigue
- The system shall estimate *worst*, *likely* and *best* cases using reliability of the underlying estimates
- The system shall request *context relevant information* from the context module
    - Possible information can be time of day and situation complexity
- The system shall *personalise* these estimations to improve accuracy (if informed consent is given)
    - Detect *driver ID*
        - Possibly request sleep/driving history and driver age via HMI
    - Estimate driver fitness with *individualized algorithms* tuned to specific drivers
    - Estimate comfort based on *historical data* on user acceptance or rejection of the takeover suggestions

---

The system shall determine the *driver state class* as fit, distracted or fatigued

---

The system shall estimate *worst*, *likely* and *best*-case *time to driver discomfort*

- The system shall compare upcoming *driving situations* with the identified uncomfortable driving situations
    - The system shall request relevant information from the *context module*
- The system shall estimate the *time* until the uncomfortable situation will occur
- The system shall estimate *worst*, *likely* and *best* case using the probability that comfort will be increased
- The system shall *personalise* the probability of a situation being uncomfortable to improve accuracy

---

Automation State

The system shall estimate *worst, likely and best-case time to automation (un)fitness* based on automation fitness estimates for the current driving context:

- Estimate the current automation fitness
    - Estimate the current automation fitness score
    - Access relevant information from the automation system to estimate the current automation fitness score
- Estimate the predicted automation fitness
    - Estimate the predicted automation fitness score
    - Access relevant external driving context information
- Estimate when the driving automation system is unfit to drive
    - The driving automation system is deemed unfit to drive if it can no longer execute its defined dynamic driving task due to degraded automation performance (low automation fitness score)
- Estimate the time to automation unfitness as the shortest time when the estimated automation fitness score becomes lower than a cut-off threshold
- Estimate the time to automation fitness as the shortest time when the estimated automation fitness score becomes greater than a cut-off threshold
- Estimate worst, likely and best-case scenarios of time to automation (un)fitness using the reliability of its inputs, both internal and external

---

The system shall determine the *active automation level* as either none, CM, SB or TtS

The system shall determine the *automation state class*, i.e., the reason for an upcoming change in automation availability

---

The system shall determine the *appropriate intervention type*, i.e., a possible way to improve the automation fitness

---

The system shall *extract and collect context relevant information* from the driving automation system to the context module

Context
The system shall extract and collect *relevant context information* required for different system functions from information source such an original vehicle and Mediator specific hardware and software third party software.
Decisions
The system shall detect situations where a Mediator system *action is required* • The system shall monitor the conditions described in Table 10 using the TTDU/F and TTAU/F • The system shall assess if conditions are met taking into account uncertainties • The system shall request input via the HMI to decrease such uncertainties
The system shall select the *appropriate action* for the detected situation • The system shall estimate the probability of success of the actions described in Table 11 • The system shall optimise action selection for driver safety and comfort, which comes down to optimizing the proposed list of KPI's in section 3.5.5 • The system shall constrain the optimisation for safety • The system shall personalise the action selection to each driver (ideas on how to implement this were given in section 3.5.4.) • The system shall take into account possible questionable validity of the detected states when selecting an action and corresponding parameters
Actions[2]
The system shall *execute* the selected action • The system shall *provide all modules with the necessary information* to execute the action.   ○ E.g., Time budget for action, necessity of action, current automation level • The system shall unambiguously *make the driver aware of current automation levels*, and therefore the driver's appropriate sense of responsibility, as consistently as possible through its primary and secondary (ambient) look and feel. • The system shall as much as possible fulfil its interaction with the driver, within modes such as preventive and corrective actions, as well as in transfers between modes, through a *single, recognizable and predictable ritual*, for quick and intuitive learning. • In case the driver indicates a different automation preference than that of DL, the system should *negotiate with the driver*. For low necessity levels a seductive negotiation between automation and human is applied, while for higher levels a persuasive negotiation is applied, or even a forced take-over (no negotiation).
The system shall perform all *HMI functions of the original vehicle HMI*
Ethics & Validation
The system shall *log* all inputs and outputs relevant to determine the identified key performance indicators
The system shall be able to receive and process *inputs from an experimenter* during validation
The system shall be able to deal with *dummy variables* during validation • Dummy variables for modules and subfunctions will need to be developed in the next stage of the project
The system shall respect the *autonomy of human beings*

Table 14: Overview of hardware related Mediator system functional requirements

[2] Use case specific functional requirements for actions are described in Paragraph 3.4.10

Functional Requirements (hardware related)
Sensors
The system shall measure heart rate
The system shall measure respiratory rate
The system shall capture face video data from different angles • facial features should be visible while driving and while performing NDRT
The system shall capture wide angle video data • NDRT involvement should be visible from video • Detection of the device that has the driver's attention should be possible
Controls
The system shall communicate with the driver by accessing his/her field of view • i.e., the controls shall be able to adjust the ambient lighting and communicate to the driver via the device on which his/her attention is focussed.
The system shall communicate to the driver via multiple different modalities. • i.e., the system shall make use of controls that provide multimodal cues, such as visual, auditory and haptic cues.

Next to functional requirements also a set of non-functional requirements was identified, shown in Table 15.

Table 15: Overview of Mediator system non-functional requirements

Non-functional Requirements
The system shall be *transparent* • Drivers shall understand the decisions of the system • Drivers shall understand their responsibility at all times during the drive
The system shall be highly *usable* • Drivers shall not need extensive training to use the system • Drivers shall prefer using a vehicle with the Mediator system than a vehicle without • The HMI shall be designed for learned, familiar and general known affordances to minimise learning effort
The system shall preserve *human autonomy*
The system shall be *feasible* • The system shall be implementable within the Mediator project duration • The system shall reasonably be implementable in real vehicles in the near future
The system shall be *robust* and thus be able to cope with errors during execution
The system shall have sufficient *maturity* and thus have a low frequency of failure by faults
The system shall be *recoverable* and thus can be returned to a functional state from a non-functional state and lost data can be recovered

The system shall be *predictable* • The system shall have accountability so that it can be proved why it operates in certain ways to avoid confusion or fear within the general public
The system shall be respectful of the *privacy* laws

7. References

Ahlstrom, C., Kircher, K., & Kircher, A. (2013). A gaze-based driver distraction warning system and its effect on visual behavior. *IEEE Transactions on Intelligent Transportation Systems*, *14*(2), 965-973.

Beggiato, M., Hartwich, F., & Krems, J. (2019). Physiological correlates of discomfort in automated driving. *Transportation Research Part F: Traffic Psychology and Behaviour, 66*, 445–458. doi:10.1016/j.trf.2019.09.018.

Borowsky, A., Oron-Gilad, T., Chasidim, H., Ahlström, C., Karlsson J.G., Bakker, B., Beggiato, M., Rauh, N., & Christoph, M. (2020). *Behaviour Markers for Degraded Human Performance*, Deliverable D1.2 of the H2020 project MEDIATOR.

Brooke, J. (1986). SUS: a "quick and dirty" usability scale". In P. W. Jordan; B. Thomas; B. A. Weerdmeester; A. L. McClelland (eds.). *Usability Evaluation in Industry*. London: Taylor and Francis

Christoph, M., Cleij, D., Ahlström, C., Bakker, B., Beggiato, M., Borowsky, A., ..., & Van Nes, C.N (2019). *Mediating between human driver and automation: state-of-the art and knowledge gaps*. D1.1 of the H2020 project MEDIATOR.

Ethics Commission (2017), *Automated and Connected Driving* - A Report Commissioned by the Federal Ministry of Transport and Digital Infrastructure, Germany.

van Grondelle, E. et al. (in preparation). *HMI functional requirements*. Deliverable D1.5 of the H2020 project MEDIATOR.

Hart, S.G., & Staveland, L.E, (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Advances in Psychology*, north-holland publishing company: Amsterdam, The Netherlands

Hergeth, S., Lorenz, L., Vilimek, R., & Krems, J.F. (2016). Keep Your Scanners Peeled: Gaze Behavior as a Measure of Automation Trust During Highly Automated Driving. *Human Factors, 58*, 509–519, doi:10.1177/0018720815625744.

Hermens, F. (2020). *Fatigue and distraction detection; A review of commercially available devices to detect fatigue and distraction in drivers*. (Report NO. R-2020-6), SWOV https://www.swov.nl/publicatie/fatigue-and-distraction-detection

Institute of Electrical and Electronics Engineers – IEEE (2016). *The IEEE Global Initiative for Ethical Considerations in Artificial Intelligence and Autonomous System*. www.standards.ieee.org

Jian J-Y, Bisantz AM, Drury CG (2000). Foundations for an empirically determined scale of trust in automated systems. *International Journal of Cognitive Ergonomics, 4*(1):53–71

Kurpiers C., Biebl B., Meija Hernandez J., & Raisch F. (2020). Mode Awareness and Automated Driving-What Is It and How Can It Be Measured. *Information, 11*, 277.

Laugwitz B., Held T., & Schrepp M. (2008). *Construction and evaluation of a user experience questionnaire*. Symp Aust HCI Usability Eng Group. https ://doi.org/10.1002/97835 27617 272.ch1

Mano, D., Berger, N., Larsson, A., Brännström, M., Knauss, A., Toffetti, A., … Christoph, M. (2021).*Quantified Markers for Degraded Automation Performance,* Deliverable D1.3 of the H2020 project MEDIATOR.

Naujoks F., Purucker C., Wiedemann K., & Marberger C. (2019). Noncritical State Transitions During Conditionally Automated Driving on German Freeways: Effects of Non–Driving Related Tasks on Takeover Time and Takeover Quality. *Human Factors, 61*(4), 596–613, doi: 10.1177/0018720818824002

Saffarian, M., De Winter, J.C.F. & Happee, R. (2012). Automated Driving: Human-Factors Issues and Design Solutions. *Proceedings of the Human Factors and Ergonomics Society, 56*(1), 2296–2300, doi: 10.1177/1071181312561483

Sütfeld, L.R., Gast, R., König, P., & Pipa, G. (2017). Using virtual reality to assess ethical decisions in road traffic scenarios: applicability of value-of-life-based models and influences of time pressure. *Frontiers in Behavioral Neuroscience, 11*, 122.

United Nations Educational, Scientific and Cultural Organization – UNESCO (2020*). Elaboration of a Recommendation on the ethics of artificial intelligence*. www.unesco.org

van der Laan, J. D., Heino, A., & De Waard, D. (1997). A simple procedure for the assessment of acceptance of advanced transport telematics. *Transportation Research Part C, Emerging Technologies, 5*(1), 1-10.

Vermunt, D. (2020). *A Markov Decision Process approach to human-autonomous driving control logic*, Master thesis

Wandtner B., Schömig, N., & Schmidt G. (2018). Effects of Non-Driving Related Task Modalities on Takeover Performance in Highly Automated Driving. *Human Factors, 60*(6), 870–81, doi: 10.1177/0018720818768199