TNO innovation for life

UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering, Mathematics & Computer Science

Adding higher-order Situation Awareness components to a Platoon Commander's Battle Management System.

Daan H. Ekkelenkamp

M.Sc. Thesis
January 2021

Supervisors:
dr. ir. M. H. P. H. van Beurden (Maurice)
dr. M. B. van Riemsdijk (Birna)

Examiners:
prof. dr. D.K.J. Heylen (Dirk)
dr. ir. P.W. de Vries (Peter)

Human Media Interaction group
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Preface

This thesis forms the final work of my MSc Interaction Technology at the University of Twente. I enjoyed my years as an I-TECH student and the projects that were part of it. The majority of these projects started with the feeling of not having a single clue where to start and ended with a moment of pride when a working result was presented. These results range from a *living painting* that freezes when you look at it, to an *emotional mirror* that visualizes your emotions when you talk to it. All these different projects required me to learn new skills and ensured that I ended up with a well-stocked digital tool belt. Furthermore, I am really glad that my master's offered me the ability to take a part of my studies abroad, which in my case was Coïmbra, Portugal. It was also in Coïmbra where my digital tool belt got the biggest expansion, of which the best example is the knowledge concerning artificial intelligence and machine learning that is used in this thesis. This combination allows me to finish my master's with the confidence to be able to create any digital project that I can come up with. And, as my Portuguese friends explained to me during their graduation: *"segredos desta cidade levo comigo pra vida"*.

This thesis could not have been completed without the help and contribution of multiple persons which I owe an acknowledgment. First of all, I would like to thank TNO for offering me an interesting project for which I could formulate this research. A special thanks goes out to dr. Maurice van Beurden as my daily supervisor at TNO. Thanks for how you were always available for questions and let me formulate my thoughts during our conversations. The same thanks goes to dr. Birna van Riemsdijk as my supervisor from the University of Twente. Thanks for the fact that you accepted the invitation as my supervisor before you even had time to settle at your new position and thank you for your patience and structured approach during the process. Furthermore, I owe a lot of thanks to prof. Dirk Heylen and dr. Pieter de Vries for their feedback that allowed me to further improve the quality of this thesis.

Thanks, Job and Sarthak, for the daily stand-ups, the push-ups, and the table tennis games. Thanks, Tatjana, Eva, Luuk, and Marit for the online morning meetings that allowed us to discuss our progress and stay connected during the times that we

weren't able to go to the office. Thanks, Henri, for being willing to meet and share your first-hand experience from the *Korps Mariniers*. Thanks, Mark, for explaining and getting me started in the Socio-Cognitive Engineering tool. Thanks Adriaan and Birgit for taking the time to read and comment on my thesis. And thanks Jules, for helping me in getting access to the reading materials that were unavailable to me.

Finally, I want to thank my family and friends for their support and encouragement in the past 10 months.

Daan H. Ekkelenkamp
Utrecht, January 2021

Summary

The goal of this study is to support Platoon Commanders (PC) in their decision-making processes by increasing their situation awareness (SA) using artificial intelligence (AI). Good SA will increase the likelihood of good decision-making performance and consists of perceiving critical information (level 1 SA), comprehending its meaning (level 2 SA), and being able to project its future status (level 3) [1]. Examples of these elements in a battle scenario are the location of a friendly section that is approaching the enemy (level 1), extracting from this location whether this section is still following the planned route (level 2), and estimating the time at which this section will reach the enemy (level 3). Recent developments at the Dutch Ministry of Defence provided Dutch commanders at platoon level with a large smartphone, called the Raptor, that increases the SA of the PC [2] [3]. The Raptor supports mainly level 1 SA and slightly level 2 SA by, among others, providing the PC with a map, current locations of friendly sections, and the most recently known locations of enemies.

This study creates an example of how AI can be used to add an interpretation of the current state of the battlefield (level 2 SA) and a projection of it in the near future (level 3 SA) to the Raptor. Including this information in the Raptor should increase the SA of the PC and therefore should improve the PC's decision-making performance even more. A lot has been written about the construct of SA, different ways to measure SA, and best practices to take SA into account during a design process. Yet, existing research does not contain a method to include particularly higher-order SA components in a system. This study proposes such a method composed of existing methods and uses it to create an example of how AI technology can be used to support a Platoon Commander.

The study focuses on the cognitively challenging scenario of the hasty attack, in which PCs have little information about the location and numbers of enemy forces that they might encounter [4]. Due to a lack of information, quickly building SA is extra important as this determines the decisions that the PC will make (e.g. what is the best approach route, how to position the sections, what weapon systems are needed for the attack) [1] [4]. At the same time, the PC is likely to be confronted with unexpected events that will increase the cognitive load even more (e.g. the terrain is different than expected, casualties occur, another enemy appears) [4]. Because of this cognitively challenging character, this study focuses on the hasty attack; it is the scenario where supporting SA can prove most helpful to the PC.

The process of adding level 2 and level 3 SA components to the Raptor starts by performing a Goal-Directed Task Analysis (GDTA) [5] to gain insight into the goals and dynamic information needs of a PC during a hasty attack. A literature review, doctrine, videos explaining military tactics, an interview with a former PC, and recordings of PCs executing a hasty attack were used to identify all the different information components that a PC uses throughout a hasty attack. Together with a former PC, this list of SA components has been evaluated and grouped into three functionalities that can be added to the Raptor: (1) Monitoring whether each section is moving according to the planned route, (2) monitoring and predicting risks of fratricide, and (3) predicting and monitoring locations where the enemy is likely to have hidden Improvised Explosive Devices (IEDs).

These functionalities were then iteratively translated into a system- and visual design aimed to increase SA. The Socio-Cognitive Engineering method [6] was used to specify the design while simultaneously focussing on usability and the required technology. A set of SA oriented design principles were used to take the best practices in supporting SA in design into account and were provided by the Design for Situation Awareness method [7]. The Heuristic Evaluation [8] was used to evaluate the initial design on its usability without the involvement of target users. The initial design was improved by overcoming the identified usability issues in an iteration.

This design was then turned into a functional prototype that was evaluated with two former PCs. The evaluation was performed online and the subjects were shown the execution of a scripted hasty attack in a virtual environment together with the display of the prototype. In the scripted hasty attack, sections were diverting from their route, walking towards predicted IEDs, and causing future- and acute risks of fratricide to show the subjects how the new functionalities respond to these situations.

The added functionalities were considered useful in practical scenarios and the subjects liked this way of information visualization because it was directly in line with how the PC thinks and works. The prototype offers the PC information in space and time variables (e.g. how does a section need to move to get back on its planned

route and how long will this take) and made the current situation understandable at a glance (level 2 SA) and indicated the impact on the rest of the mission (level 3 SA). The IED predicting functionally received the most feedback as although this information was considered to be useful, the subjects expected this functionality to reduce the vigilance of the PC for IEDs in other locations. Furthermore, both subjects also indicated that they would prefer a functionality like this to indicate key locations (high buildings, locations offering good lines of sight and fire) instead of IEDs, which shows that broadening of this functionality is needed. In general, the results from the evaluation show enough reasons to continue the development of the created functionalities and suggest that the proposed method is suitable to integrate higher-order SA requirements in an existing system. .

Contents

List of acronyms

PC	Platoon Commander
SA	Situation awareness
AI	Artificial intelligence
GDTA	Goal-Directed Task Analysis
IED	Improvised explosive device
SU	Situation Understanding
SCE	Socio-Cognitive Engineering
HE	Heuristic Evaluation
MARS	Mission Awareness Rating Scale
SME	Subject matter expert
SQC	Squadron Commander
WDA	weapon danger area
MCC	Mission critical cue

Chapter 1

Introduction

Imagine a scenario where a platoon of soldiers was marching through an urban environment and just made contact with enemy forces. The front section that made the contact took cover and became the fire support section as they are now returning precision fire to suppress and pin down the enemy. The shots and bullet impacts create a lot of noise and the attack induces high stress levels on the entire platoon. The Platoon Commander (PC) quickly assesses the situation using his tactical handheld display called the Raptor and develops a plan to attack the enemy while keeping his own sections safe. In order to aid the PC in assessing the situation, the Raptor shows four predicted locations of improvised explosive devices. The PC uses this information to create a plan of attack that avoids the explosives, sees the opportunity to surprise the enemy with a flanking movement, and communicates the plan to the rest of the platoon. The PC updates his own superior about the current situation of the platoon, and quickly thereafter receives an audio alert from the Raptor. The flanking section made a navigation mistake, diverted from their planned route, and will now need an extra minute to reach the enemy. The PC quickly contacts the fire support section and commands them to reduce their fire rate since they need to be able to suppress the enemy for an extra minute and need to save their ammunition to do so. The PC moves to a safer location to reduce the risk of getting shot. Just when the PC enters a building the Raptor sends another audio alert and warns the PC about a potential risk of friendly fire as the flanking section is moving towards the line of fire of the fire support section. The PC looks at the Raptor and sees that the flanking section is 30 seconds removed from the enemy. The PC commands the fire support section to slowly divert their fire and communicates that the flanking section will arrive in 20 seconds. The fire support section diverts their fire in time and the flanking section surprises and defeats the enemy.

Although the above mentioned Raptor sounds like a useful tool for PCs, the current reality is that PCs still have to plan attacks using just a map and compass, and that their only electronic aid consists of a portable radio. For a successful execution of the mission, it is important for the PC to correctly understand the battlefield, which is described in scientific literature by a construct called situation awareness (SA). Good SA consists of perceiving critical information (level 1 SA), comprehending its meaning (level 2 SA), and being able to project its future status (level 3) and will increase the likelihood of good decision-making performance [1]. Examples of these elements in a battle scenario are the location of a friendly section that is approaching the enemy (level 1), deriving from this location whether this section is still moving according to the plan (level 2), and estimating the time at which this section will reach the enemy (level 3). The military term for the kind of mission mentioned in the scenario above is the hasty attack and is an example of a mission in which it is particularly difficult for a PC to obtain good SA because it is physically- and cognitively demanding and characterized by uncertainty (e.g. limited terrain knowledge and unknown threat location) [4].

The scenario at the same time shows how technology might be applied in the future to support the PC in commanding his platoon. The recently expected introduction of an initial version of the Raptor should make a start in doing so and will support a PC in his SA [3] [2]. It is part of recent developments of the Dutch Ministry of Defence and consists of a large mobile phone that is carried on the chest (see figure 1.1). This initial version will not include the automated functionalities as mentioned in the scenario, but will mainly display level 1 and some level 2 SA information (e.g. a map, location of each section, the last known locations of enemies, and their relative positions). However, recent and past developments in technology allow for the development of supportive technologies that could include the automated functionalities in the Raptor that are mentioned in the scenario above. Plenty of research is - and has been - focussing on artificial intelligence (AI) that can interpret large datasets and use them to make predictions. However, less research has been performed on what kind of information this technology should produce to be useful for a PC and how it should be presented.

The goal of this study is to use AI to make an interpretation (level 2 SA) and projection (level 3 SA) of certain elements on the battlefield and incorporate these in the Raptor like sketched in the opening scenario and illustrated in figure 1.2. In doing so the study will examine what kind of information AI needs to generate to support the PC, how this should be presented, and whether a PC will indeed experience the resulting solution to be useful. There currently exists no research methodology that is specifically aimed at creating these higher-order SA components, which requires a focus on SA, AI and human factors at the same time. This study combines a set of

Figure 1.1: *The Raptor in context. Taken from [3]*

existing methodologies into a composite methodology in order to add these higher-order SA components to the Raptor.

This process will start with an analysis phase to break down the dynamic information needs of the PC throughout a hasty attack to identify all relevant SA components (chapter 3). A former PC is then consulted to verify these SA components and to combine the SA elements which he believes will be most valuable for the PC into functionalities that can be added to the Raptor (Chapter 4). Once these functionalities have been defined, they will be specified in more detail and turned into a system- and visual design (chapter 5). Finally, this design will be realized into a prototype that can be used to evaluate the design with experienced PCs (chapter 6). The evaluation of this prototype is discussed in chapter 7 and the entire process is discussed and concluded in chapter 8. All these different steps should answer the research questions that have been formulated:

- RQ 1: How can the current version of the Raptor be enhanced with higher-order SA components to increase the SA of the PC during a hasty attack?
- RQ 1.1: Which information is important in building the SA of the PC during a hasty attack?

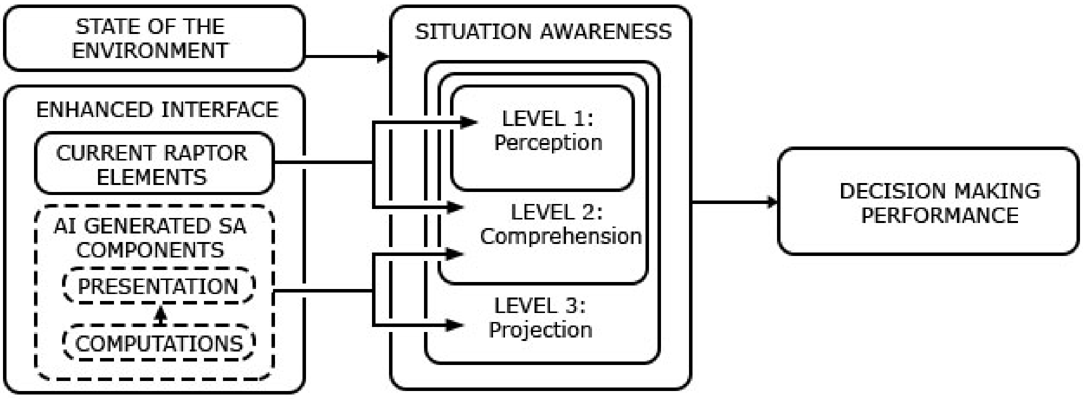


Figure 1.2: *Visualization of the goal of this study that aims to include the elements embedded by the dashed line in the Raptor*

- RQ 1.2: Which important SA components are not already included in the current version of the Raptor and are technically realizable within the timespan of the research?
- RQ1.3: How can the computed output be presented to the PC in an effective manner?
- RQ1.4: How can the created design be realized into a testable prototype?

This study will contribute to addressing broader challenges of incorporating automated technologies like AI in human tasks [9]. Transparency in AI is expected to reinforce and facilitate collaboration between humans and AI [10] and methods have already been proposed to achieve this by letting AI explain its answers [10] [11]. This project aims to optimally combine the strengths of both humans (e.g. cognitive capabilities like planning, creativity and problem solving) and machines (e.g. numerical computations, statistical reasoning and information retrieval capabilities) [12] to enhance the intelligence of the Raptor and support the PC in his SA. Multiple design methods exist that focus on such a collaboration between humans and AI and the interdependence between the two. One example that is used in this study is the Socio-Cognitive Engineering method that is specifically created to combine AI and human factors [13]. This project aims to particularly use AI that *"increases the capability of a man to approach a complex situation, to gain comprehension to suit his particular needs, and to derive solutions to problems"* [14] and lets the PC stay in control of making the decisions.

Chapter 2

Background

This chapter introduces the background that is related to this study. First of all, the context of the project and the military context that is used in this project are introduced. After this, three related topics are discussed: situation awareness, human decision making, and human-machine cooperation.

2.1 Context

This study is executed as part of a larger project at TNO, called the Decision Support project, and serves as a master thesis for the Interaction Technology master of the University of Twente. The Decision Support project explores where in the decision-making process artificial intelligence (AI) technology can augment the performance of a future Platoon Commander (PC). The Decision Support project is particularly focussing on how AI technology can be embodied in the equipment of a soldier and how a future soldier will interact with the technology. The study presented in this document provides the Decision Support project with an example of such an embodiment and shows what kind of information AI needs to create to support PCs in their situation awareness (SA), how this should be presented, and whether PCs will indeed experience the resulting solution to be useful.

This study focuses on the role of a PC, which is the commander of a platoon (often consisting of three sections, including eight men each). In this document, the groups lower in the hierarchy are referred to as sections and multiple platoons are referred to as a squadron. The PC gives commands to the sections via the Section Commander and reports to- and receives orders from the Squadron Commander. This hierarchy is also illustrated in figure 2.1.

The context of this study is the cognitively challenging scenario of a hasty attack. A

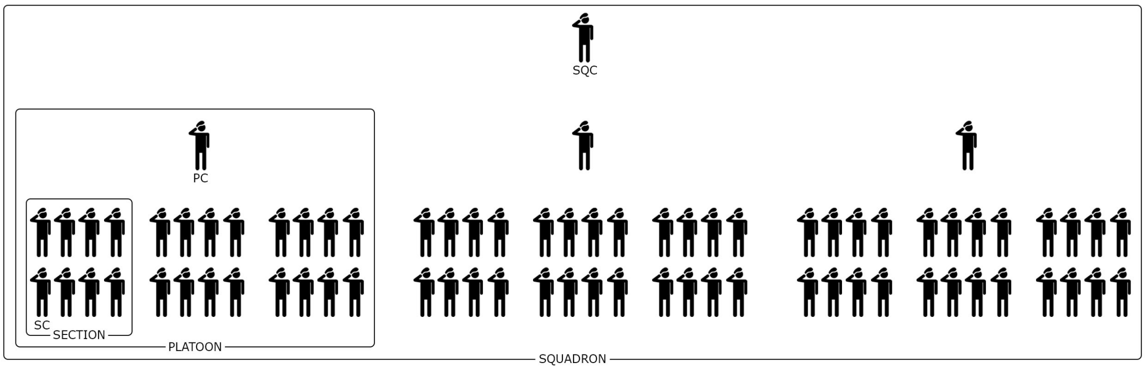


Figure 2.1: *Overview of military hierarchy surrounding the PC*

hasty attack and is characterized by uncertainty (e.g. limited terrain knowledge and unknown threat location) and limited preparation time [4]. Typically for a hasty attack is that enemy presence is expected, but information concerning the location and strength of the enemy is missing. When the platoon makes contact with the enemy it is the task of the PC to lead the platoon. First of all, the PC needs to determine whether the platoon will attack the enemy, ask for back-up, or retreat. When the platoon is going to attack, the PC needs to come up with a plan of attack on the spot. The section that made contact with the enemy will take the enemy under fire and keep them suppressed and pinned down. An often selected approach is to surprise the enemy with a flanking movement by another section, but whenever the terrain or a time limit prevents this, a frontal approach is also an option. Once the PC made and communicated a plan of attack, the PC has to make sure that the execution of the plan is going smooth and that the platoon stays safe. To do so, the PC closely monitors the execution of the attack, deals with unexpected events like casualties, and stays in contact with the squadron commander. The intense nature of the hasty attack induces high-stress levels and time pressure on the PC [4]. Because of this cognitively challenging character, this study focuses on the hasty attack; it is the scenario in which the PC can use support in his decision making.

2.2 Situation awareness

As mentioned before, this project aims to increase the decision-making performance of PCs by increasing their SA. Endsley claims that SA is a crucial construct on which decision-making and performance hinge [7]. An increase in SA is therefore likely to increase decision-making performance. Endsley defined SA as *"the perception of the elements in the environment within a volume of time and space, the comprehen-*

sion of their meaning, and the projection of their status in the near future" [1]. A PC needs to have good SA as this knowledge will form the basis for the decisions that the PC makes, the plan that the PC creates, and the way the PC monitors the execution of the attack. Moreover, making the correct decisions is ever more important in a field where bad decisions can lead to a loss of life [15]. Therefore SA is very relevant in the military domain and it is no coincidence that the study that defines the concept of SA concerns military personnel: fighter pilots [1].

As follows from the definition of SA that has been presented, true SA is more than just being aware of the numerous pieces of data (like the location of friendly troops for example). The definition makes a clear distinction between the perception of elements, the comprehension of their meaning, and the projection of their status in the near future. To make these differences clearer, Endsley has divided them into three levels [5].

The first level of SA encompasses the perception of the status, attributes, and dynamics of relevant elements in the environment [1]. Examples of elements in which the PC is interested are the location of troops, level of ammunition, active weapon system, dynamics of both friendly and enemy forces in a given area, and their relationship to other points of reference [4].

The second level of SA focuses on the comprehension of the situation. This comprehension builds upon the synthesis of the separate Level 1 elements [1]. In level 2 SA the operator is not only aware of the elements that are present, but he also understands their significance in the light of his goals [1]. Together with the level 1 elements, especially when these elements are put together to form patterns or gestalts with other elements, the operator creates a holistic image of the environment, understanding the significance and relevance of objects and events [1]. The PC must understand that the appearance of enemy troops within a certain proximity of each other and in a certain location will indicate things about their objectives, like whether they are likely to strike an attack or not. Unlike level 1 SA, a novice might not achieve the same level 2 SA when compared to an expert, since a novice might lack the ability to integrate various data elements along with his goals to comprehend the situation [1].

The highest level of SA is formed by the ability to project the future actions of the elements in the environment. This projection is achieved through knowledge of the status and dynamics of the perceived elements and the comprehension of the situation [1]. An example of level 3 SA for the PC is that he can determine whether there is enough time available to make a flanking moving or not. This knowledge allows the PC to decide on the best course of action to meet his objectives [1].

It is good to note that in the military domain Situation Understanding (SU) is often-used besides SA. SU is more concerned with what the received information means in respect to mission threats, mission opportunities, and information gaps based on the obtained SA [16]. For that reason, there is a great overlap between level 2 SA and SU. This approach of putting information in a personal perspective can also be seen in the sensemaking model of Klein [17] that, among other things, states that different people viewing the same events can perceive and recall different things depending on their goals and experiences. In the case of military personnel, their training and mission goals will highly influence their understanding of the situation based on their SA. This project will focus on SA and does not treat SU and sense-making in further detail. The reason for this lies in the fact that SA can be seen as a requirement for SU and sensemaking and therefore describes the processes on a more global level. Furthermore it is also the most widely accepted construct of the three.

It is also important to understand the difference between situation awareness and situation assessment. The former relates to a state of knowledge of which we have seen the definition and the different levels. The latter are the processes used to achieve this state [1]. As a result, SA is presented separately from other cognitive constructs such as attention, working memory, workload, and stress. These constructs are all related and they can affect SA, but they are considered as separate constructs [1].

2.3 Human Decision Making

One cognitive construct that is closely related to SA and of high relevance in a hasty attack is cognitive load. High cognitive load has been proven to reduces the performance of complex tasks [18]. When experiencing high cognitive load, humans tend to automatically adopt strategies to cope with this high load. Examples of these are resistance to considering alternative hypotheses (cognitive tunnel vision) [18] [19], using known knowledge even though the situation requires different knowledge (assimilation paradox) [19], and narrowing of attention which causes the subject to miss environmental cues (temporal narrowing) [18] [19]. The amount of information that military commanders and staff must process has increased tremendously, while the amount of time available for decision-making has decreased dramatically [20].

The Canadian Army even started researching the potential cognitive overload that soldiers can experience due to technological developments [21]. Therefore, further development and incorporation of technological aids should support soldiers without posing an additional cognitive load.

2.4 Human-Machine cooperation

Incorporating automated technologies like AI in human tasks has been shown to be rather complex [9]. It turns out that humans tend to be somewhat reserved in believing and using outcomes of automated technologies [10]. Achieving transparency in AI is expected to reinforce and facilitate the collaboration between humans beings and AI [10]. Methods have already been developed to have more transparent AI by letting it explain its answers [10] [11]. Besides academics, also governmental institutions have been proclaiming the necessity to make the interaction between humans and machines safe. The European Commission's High-Level Expert sets the foundation of trustworthy AI in fundamental rights by four principles: *respect for human authority, prevention of harm, fairness, and explicability*. Furthermore, according to Jonker and colleagues, common ground and mutual predictability are considered to be important for effective coordination in human-machine collaboration. Jonker and colleagues believe that *"shared understanding between humans and machines"* is one of the greatest challenge that developers of human-machine interactions face [22].

Both machines and humans have their strengths and weaknesses. Machines are very effective in numerical computation, information retrieval, statistical reasoning, and (can) have almost unlimited storage. Humans, on the other hand, have cognitive capabilities which include consciousness, problem-solving, learning, planning, reasoning, creativity, and perception. This enables humans to learn from past experiences and to use this experiential knowledge to adapt to new situations and to handle abstract ideas to change their environments. The combination of both humans and machines can be used to enhance the intelligence of systems and support the PC [12].

Multiple design methods have been developed that focus on the collaboration between humans and AI and the interdependence between them. One example is the co-active design framework that creates an overview of tasks and can help to identify which team member needs support in certain tasks while at the same time showing which team member can offer this support [23]. Another example is called Socio-Cognitive Engineering and emerged from the field of cognitive science and AI and is specifically created to combine AI and human factors [13].

This project aims to use a particular kind of AI to optimally combine the strengths of both humans and machines. The kind of AI that is used should be focusing on aiding the PC in making decisions and should not make the decisions for him. This project aims to use AI that *“increases the capability of a man to approach a complex situation, to gain comprehension to suit his particular needs, and to derive solutions to problems”* as defined by Engelbart [14].

Chapter 3

Methodology

3.1 Introduction

This chapter globally introduces the methodology that has been created to answer the main research question of this project: *"How can the current version of the Raptor be enhanced with higher-order SA components to increase the situation awareness (SA) of the Platoon Commander during a hasty attack?"*. The most important characteristic of the methodology that has been created is that it had to focus on multiple areas at the same time. The method mainly focusses on increasing situation awareness (SA). At the same time, the method had to be able to deal with a challenging technology part to include the artificial intelligence (AI) technology that was envisioned as a way to compute the higher-order SA information. Finally, the method also needed to focus on usability, or in a broader sense, human factors. The result will be used in a very stressful situation, in which there is no room for unusable interfaces as mistake can results in the loss of life [15]. To focus on all these three elements at the same time a methodology has been composed of existing methods. In this chapter, the main concepts behind the used methods and how they relate to each other are discussed. First, all the individual methods are explained in detail after which the composite method is presented which has been used for this project and is created by combining the three existing methods.

The following existing methods were combined into a single method for this project, and visualized in figure 3.2:

1. Socio-Cognitive Engineering (SCE), which combines a classical human-centered perspective with a technology-centered perspective [6].

2. Designing for Situation Awareness (D4SA), which provides a methodology and design principles to create systems aimed at increasing SA [7].

3. Heuristic Evaluation (HE), which is used to evaluate the usability of the initial design without a subject matter expert. [24].

3.1.1 Socio-Cognitive Engineering

The SCE methodology does not only allow to verify that the technology is performing in a technically reliable manner, but also makes sure that the technology successfully achieves its objectives with respect to the common goal in an effective and efficient manner. At the same time the method makes sure that this technology is easy and intuitive to use [6].

The key to the methodology is the generation, refinement, validation, maintenance, and reuse of coherent and concise design specifications. These specifications describe both what the technology should do and the underlying design rationale (the why and when).

The SCE process consists of three parts: (1) the foundation-, (2) the specification- and (3) the evaluation phase.

1. **The foundation** consists of three elements: operational demands, the envisioned technology, and relevant human factor knowledge. Together these elements describe the problem to be solved, the existing knowledge to solve this problem, and the technology to implement that solution.

   Operational demands The operational demands consist of the analysis of the problem description and the stakeholders. How this analysis is performed is not structurally specified in the method. The analysis method for this project is selected from the literature base on situation awareness.

   Envisioned technology The envisioned technology describes the available options of existing technology that can be used to come to a system solution. In this project the envisioned technology is AI and the specific kind of AI that is envisioned is already discussed in section 2.4

   Human factors The human factors knowledge mainly concerns existing best practices of how technology can be designed such that the user can work with the technology. This is, again, specific to each project and should be relevant for the problem and the chosen design solution.

2. **The specification** phase describes the solution of the problem in the form of a system design that uses the relevant human-factor knowledge and the envisioned technology that was collected in the foundation phase. This phase consists of three parts again: use-cases, functional requirements, and claims. Since these steps are well defined within the SCE method, they are explained in more detail.

   Use-cases
   A use-case consists of a specific description of step-by-step interactions between the system and the operator (e.g. Raptor and the Platoon Commander (PC)). use-cases are created for all the different scenarios in which the envisioned functionalities can be used. Within a use-case, alternative steps are presented as well. This is done to explore all the possibilities that might occur during a single use-case and design the functionalities of the interface in such a way that they respond adequately in all use-cases. An example of a use-case within this project would be a PC monitoring the execution of an attack. The steps of the use-case could then describe a PC looking at his Raptor and seeing the sections executing the attack as planned. Alternative steps could describe how a flanking section moves too far forward and ends up behind the enemy, or how the fire support section accidentally causes a risk of fratricide. use-cases make the design more concrete by describing exactly how the technology should respond in all the different use-cases. Together, the use-cases provide a detailed description of the interactions between the technology and its user in a wide variety of scenarios.

   A use-case is documented as a list of steps that describe the use-case in detail. The circumstance of the use-case is specified together with a precondition that describes the situation at the start of the use-case and a postcondition that describes the situation after the use-case to place the use-case more in context.

   Functional requirements
   Some steps of a use-case require the technology to be able to perform certain actions or obtain certain capacities. These are collected and called functional requirements. These requirements form a list of specific abilities that the technology should provide to its user. For each requirement, it is specified from which use-cases this requirement originates.

   Claims

The SCE methodology links the system's functional requirements, the system's objectives and the hypothesis to be tested during an evaluation of the system. Claims can be seen as the underlying objectives of all functional requirements. By linking requirements to claims, the designer needs to formulate hypotheses that need to be tested during the evaluation of the system. Claims should be measurable effects of the system. If a claim cannot be proven during a system evaluation, the designer needs to refine the system design, for instance, by improving or replacing the functionality. There is no use in including a functionality of which the underlying claim is not achieved.

3. **The evaluation** phase makes up the last part of the SCE method. Design evaluation is used to validate the systems' design or to identify flaws in the design that can be improved in incremental development cycles. This phase specifies which artefact is used to perform the evaluation, which method is used to perform the evaluation, and discusses the evaluation results.

   Artefact The artefact describes the embodiment in which the envisioned design will be put to the test. This can differ from a paper prototype to a fully functional system. The artefact should incorporate a given set of requirements and technological means.

   Evaluation method The evaluation method describes the method by which the evaluation of the artefact is performed. The method can take many forms such as a human-in-the-loop study, use-case simulation, or an expert interview.

   Evaluation results The outcomes of the evaluation are described in the evaluation results. Since the SCE method consists of iterative and rapid research cycles, the evaluation does not necessarily include all specified requirements, claims, and use-cases that were created in the specification phase.

3.1.2 Design for Situation Awareness

In general, the D4SA methodology describes a user-centered design approach that contains a lot of elements that are focussing on improving SA. The D4SA method itself does not contain a design method of its own but refers to a couple of widely known methods like the waterfall- and agile models. It contains a method to identify all the SA requirements that are used by an operator in a specific context (e.g. a PC during a hasty attack), different SA-oriented design principles, and different tools to measure SA.

Goal-Directed Task Analysis

The analysis part of the D4SA method is called the Goal-Directed Task Analysis (GDTA). This analysis is developed to identify all the information elements that are used to build SA during a specific task. Endsley refers to these information elements as SA requirements since they are required in building SA. The purpose of the GDTA is to document what kinds of information operators (PCs in our case) need to perform their job. At the same time, it describes how this information is used to make a particular decision. The GDTA focuses on the (dynamically changing) goals of the operator, the decisions that need to be made to obtain these goals, and the SA requirements on which these decisions are based. These are then structured in an overview as shown in figure 3.1. The resulting knowledge enables designers to create better ways to present information to operators to support SA, and consequently, decision making and performance [5] [7].

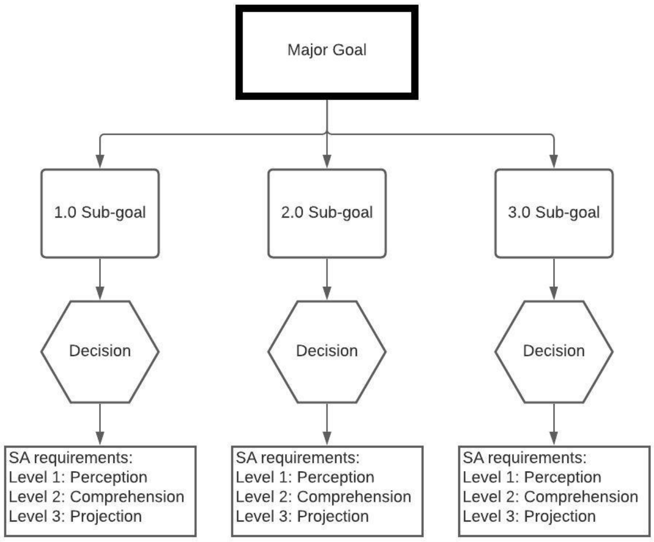


Figure 3.1: *Example of the structure of the goal hierarchy that is the result of the GDTA*

The GDTA focuses on what an operator ideally would like to know. In doing so it does not consider the way this information is conceived, as this may also vary from person to person, from time to time, from system to system, and with advances in technology [7]. This means that the GDTA can be used to identify information that the operator would like to know, but is not available in the current system. This also means that the analysis is not limited by the current technological limitations; basing the SA requirements only on current technology would induce an artificial ceiling effect and would prevent the researcher from finding much of the information that the operator ideally would like to know, and which can be used in the design efforts.

How the GDTA is constructed is not bound to a fixed set of means. However, Endsley does suggest that interviews with subject matter experts are an indispensable source for information gathering for the GDTA. On a more general scale, Endsley suggests that the researcher gets familiar with the domain and at hand and starts by gathering information about the goals that the operators at hand are seeking to achieve while performing their job. Unstructured interviews with subject matter experts are suggested for this phase. The researcher then has the task of organizing all the information that was collected into a workable preliminary goal structure that will allow for adequate portrayal of the SA requirements. Organizing pieces of information into similar categories is suggested to help in this process. Once this draft goal structure is created, it can be used in future interview sessions to refine and supplement it.

Goals are defined as higher-order objectives that are essential to successfully perform the task at hand. Goals should be descriptive enough to explain the nature of the subsequent branch and broad enough to encompass all elements related to the goal being described. The decisions that are needed to meet each goal specified in the goal hierarchy are listed beneath the goals to which they correspond. Decisions are posed in the form of questions. Important to note is that questions that can be answered with yes/no are not qualified as decisions. When a question's only purpose is to discern a single piece of information it should be qualified as an SA requirement. SA requirements present all the information that an operator needs to make a decision listed in the goal hierarchy. SA requirements can be all kinds of information and one piece of information, for example, the position of a friendly section (level 1 SA), can be used assess the deviation from the assigned route of this section (level 2 SA) as well as the expected time of arrival of this section at a certain waypoint (level 3 SA).

SA oriented design principles

The D4SA method also contains a set of design principles for engineers and designers who are seeking to *"nourish the SA of their system's users"* [7]. The principles support the process of creating system interfaces that are effective at creating a high level of SA. The principles were developed based on best practices of what is known to date on the mechanisms, strengths, and weaknesses of human SA. In total, the list of SA-oriented design principles consists of a list of 50 principles, which can be found in appendix A. These 50 principles are grouped in different areas being general-, certainty-, complexity-, alarm-, automation-, and multi operator design principles. An example of one of these guidelines is number 31 that states *“Minimize alarm disruptions to ongoing activities”*.

The methodology states that the reason why building good SA is so difficult can be related to both features of the human information processing system and features of complex domains that interact to form SA pitfalls [7]. These SA pitfalls are factors that prevent or undermine SA in many systems and environments. Recognizing these pitfalls is an important step to start designing for SA. The eight SA pitfalls are considered to be:

1. Attentional Tunneling: fixating on a single set of information to the exclusion of others.
2. Requisite Memory Trap: relying on limited short-term memory.
3. Workload, Anxiety, Fatigue, and Other Stressors: reducing a person’s capacity to process information
4. Data overload: overwhelming amounts of data can reduce SA.
5. Misplaced salience: drawing attention away from important information.
6. Complexity Creep: systems with too many features make it difficult for a person to develop an accurate mental model of how the system works.
7. Errant Mental Models: use of wrong mental models leads to misinterpretation of information.
8. Out-of-the-loop Syndrome: Automation can undermine SA

SA measurement tools

Finally, the D4SA method also contains a list of different ways to measure the SA of an operator. The different ways to measure SA can be divided into four classes: process measures, direct measures, behavioral measures, and performance measures. In general, it is believed that direct and objective measures are the best way to evaluate a system design with respect to SA. Process measures include eye movements, communications, and verbalizations. Direct measures include objective measures such as on-line probes, *“freeze”* probes, and subjective measures based on self and observer ratings. Behavioral measures involve inferring SA from specific behaviors on specific subtasks, such as *“time to make a response (verbal or non-verbal) to some event, and correct or incorrect SA as identified from soldier verbalizations and appropriateness of given behavior for a particular situation”* [25]

For this project, the Mission Awareness Rating Scale (MARS) is used to measure SA. This is a direct and subjective measure method, that is specifically developed to measure SA among platoon leaders in training scenarios, and its use in a virtual environment has been validated [26]. It consists of two sets of four questions. The first set assesses the level of SA of the participants while the second set assesses the mental workload that was required to create that SA. The first three of the four questions of each set address the three levels of SA as defined by Endsley [1]: perception, comprehension, and projection. The fourth question deals with how well mission goals can be identified. This particular SA measurement tool has been selected because of the good fit with the project (it has been developed around the PC in a training scenario and is suitable for a virtual environment), it is easily executable and it is not interrupting the task of the subject like objective measurements would do.

3.1.3 Heuristic Evaluation

The HE is used to evaluate an initial design of the interface early in the development process and without the help of an subject matter expert. The power of the HE lies in the small number of usability principles with which it can detect the majority of the usability problems. The HE is not going to result in a design in which no usability issues are present anymore. However, by already fixing the most apparent issues the evaluation with the PC can be more focussed on the effects on SA, the usability in a practical scenario, and problems that are less apparent and require domain-specific knowledge to identify.
The nine principles that the HE uses are the following:

1. Simple and natural dialogue
2. Speak the user’s language
3. Minimize user memory load
4. Be consistent
5. Provide feedback
6. Provide clearly marked exits
7. Provide shortcuts

8. Good error messages

9. Prevent errors

The heuristics are explained in more detail in appendix B. According to the study that presents the method, *"the number of usability results found by aggregates of evaluators grows rapidly in the interval from one to five evaluators but reaches the point of diminishing returns around the point of ten evaluators"* [24]. According to the same paper, this point is reached with fewer participants when only evaluators with a background in human factors and interface design are used [8].

3.2 The composite method

The composite method combines the SCE, D4SA, and HE method and uses the SCE methodology as the main framework. The reason for this is that the SCE method has been developed as a design methodology for complex, intelligent, and interactive technology. By itself, it is already providing a focus on both usability and technology. Because of this wide scope, the SCE lacks subject-specific knowledge and methods to be applied to this project. The SA specific knowledge is provided by the D4SA method which is used in multiple stages of the SCE framework. D4SA specifies methodologies to define the SA-related operational demands and human factor knowledge and provides a method to measure the PC's SA which is used for evaluating the final solution. The HE methodology is added to perform a first evaluation of the design without the involvement of a target user. This is necessary since the availability of PCs for this research is limited.

As mentioned, the process of the composite methodology mainly consists of the SCE method as can be seen in figure 3.2. This project therefore also follows the three phases as they are indicated in this method: foundation, specification, and evaluation.

Foundation

The project starts with the foundation phase in which all the SA elements that a PC consults during a hasty attack are identified (chapter 4). The SA requirements that are the most relevant to this project are selected and combined into functionalities so they can be added to the Raptor (chapter 5).

Specification

A design process is executed in the specification phase in which the system- and visual design of the envisioned functionalities are created (chapter 6). This phase will be executed twice, as the initial design created in chapter 6 will be evaluated, after which the specification phase will be executed again to improve the initial design.

Evaluation

This phase is performed twice as well. The first evaluation will focus on the usability of the initial design. The second evaluation uses the improved design and focuses on the SA of the PC and the added value of the system in a practical scenario. For this evaluation, a prototype of the design is made (chapter 7) which is used to perform a user-test with a former PC (chapter 8).

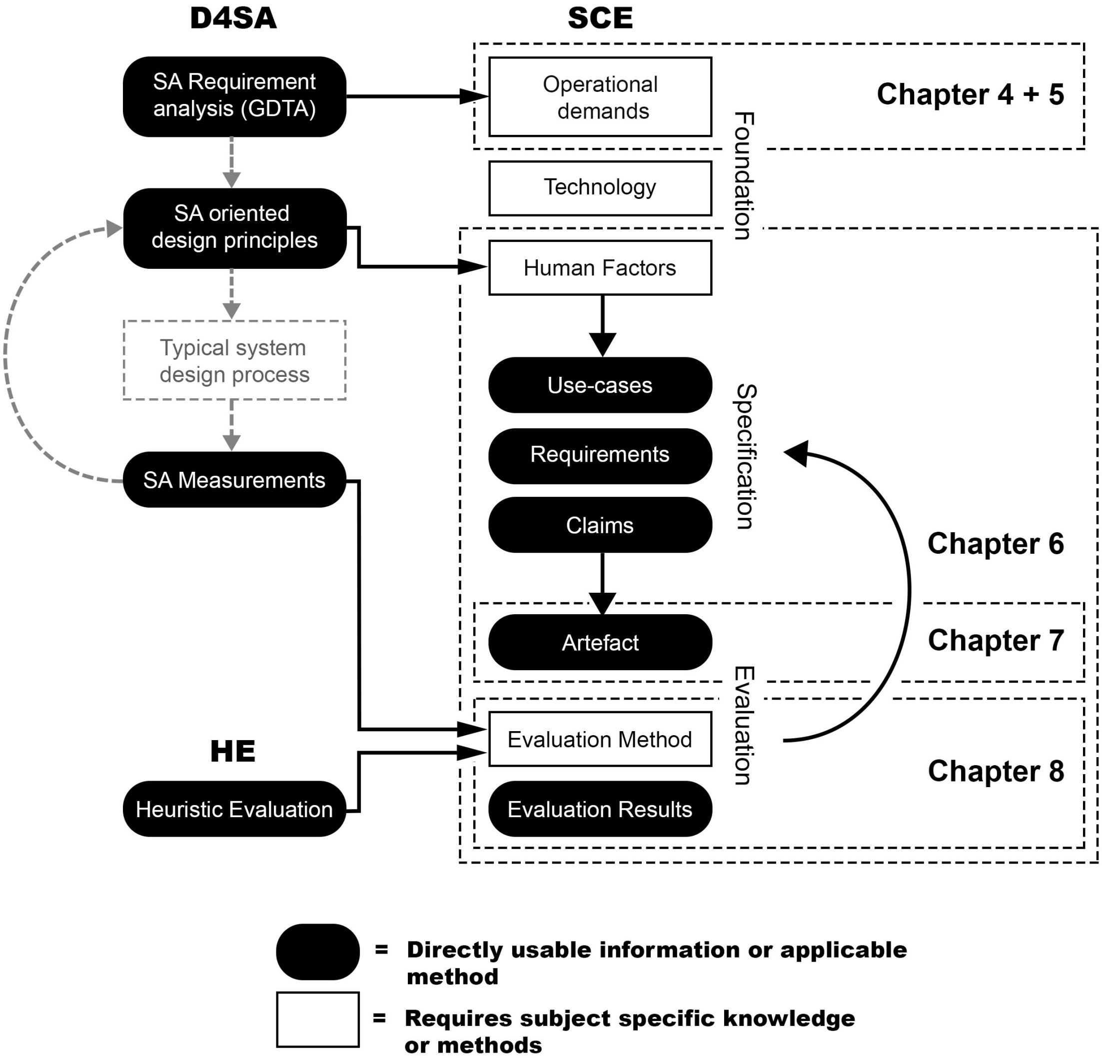


Figure 3.2: *Overview of how the used methods together form the methodology for this project*

Chapter 4

Identifying situation awareness requirements for the Platoon Commander during a hasty attack

4.1 Introduction

This chapter is focusing on specifying the operational demands in the composite method as presented in chapter 3. The operational demands describe the analysis of the problem description and the main stakeholders that are involved. In the case of this project, the problem description is presented as higher-order situation awareness (SA) elements that need to be added to the Raptor. In order to order to accurately define this problem description, specific SA elements need to be specified that will be added. And in order to be able to select these specific elements, all the SA elements that the Platoon Commander (PC) uses during

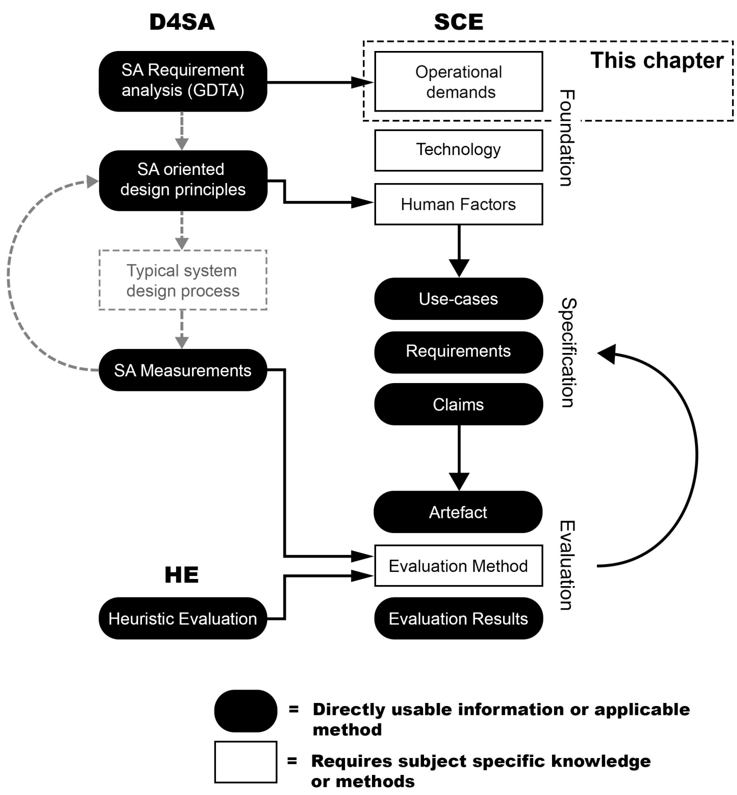


Figure 4.1: *Focus of this chapter with respect to the composite method*

a hasty-attack need to be identified in the first place. Research question RQ 1.1: *"Which information is important in building the SA of the PC during a hasty attack"* is used to define a list of all the SA elements that the PC consults during the hasty attack from which specific elements can be selected. The answer to RQ1.1, the list of all the SA requirements of the PC during a hasty attack, is created in this chapter.

To answer the research question as structured and completely as possible, a part of the D4SA method was used that is proposed by Endsley [7] [5], called the Goal-Directed Task Analysis (GDTA). This method identifies SA requirements by breaking down the goals of the PC during a hasty attack. The major goal of executing a successful hasty attack is broken down into several subgoals, decisions and finally SA requirements that are needed to achieve this major goal.

The goal of the GDTA is to create an understanding of the goals and dynamic information needs of the operation in question. Formulating the list of SA requirements should make sure that the designer/researcher obtains a clear understanding of what *'supporting SA'* means in the situation and domain at hand. In order to create this understanding, a GDTA normally includes a lot of involvement of the operator in question. Since the availability of the PC is limited for this research. Other ways of gathering the necessary information were therefore used, consisting of:

- Literature review
- Tactical material review (Doctrine and videos explaining military tactics)
- Evaluation of a preliminary goal structure with colleagues
- Evaluation of a preliminary goal structure with a former PC
- Observations of PCs leading a hasty attack

The GDTA started with a phase in which information is gathered and the goals and considerations of a PC were tried to be understood as well as possible. This information was used to create a preliminary goals structure, which was evaluated twice. At first with colleagues that have performed research on the hasty attack and the PC as well in order to find the identify the more general gaps in the goal hierarchy. Secondly with a former PC and therefore also included first hand experience. After this last evaluation, recordings of the execution of a hasty attack by PCs have been observed in order to place the comments of the PC in context and check the goal structure a last time.

The result of this chapter serves as the answer to RQ1.1 and consists of a list with all the unique SA requirements that have been identified in the GDTA. The final goal structure can be seen in appendix C. In the next chapter certain elements from this list will be selected and combined into functionalities that can be added to the Raptor.

4.2 Method

The Goal-Directed Task Analysis was used to create a list of all the SA requirements that a PC uses during a hasty attack. The method already has been presented in 3.1.2. Normally this process would consist of a lot of user involvement. For this project however, the involvement of the PC was only limited to a single interview with a PC. Indirect information sources were used to create a preliminary goal hierarchy which was then validated and supplemented in the interview with the GDTA. In chronological order the following resources were consulted for the entire process, which will be discussed in more detail in the rest of this section:

- Three related internal research reports
- Dutch Military Land Operations Doctrine Publication
- Videos explaining military tactics regarding the hasty attack
- A peer review with colleagues of the preliminary goal hierarchy
- An interview with a former PC
- Videos of PCs executing a hasty attack in a mixed reality training environment

4.2.1 Literature review

Existing research has been done within TNO with respect to either the PC, the hasty attack or the information needs of a commander in general. Three reports of related previously executed researches were examined in the literature review. This subsection will treat in which way these papers contribute to the GDTA.

A - Informatiebehoefte van de uitgestegen militair en zijn groep (The information need of the dismounted soldier and his group)

This report concerns the information needs of the dismounted soldier and his group [27] and is therefore mainly used to collect the SA requirements in the goal hierarchy.

The document includes a more elaborated literature review and includes knowledge from subject matter experts (SMEs). Although this report has the same goal, be it for a different setting, the GDTA methodology is not used as the method to obtain all the information needs of a dismounted group. The list of information components is directly formulated based on the used sources. The resulting list is very extensive but also includes a lot of information components that are not related to the hasty attack. The most useful elements were listed in a specific category, called *"tactical insight"* that mainly lists concrete and measurable information components that relate to higher-order SA requirements.

B - Commander's Dashboard - future decision support for dismounted group commanders

The goal of the Commander's Dashboard was to explore innovative concepts that support the commander of small units (between four and eight men), visualize these elements in an interface, and review how this interface was used [28]. The report includes a workshop with end-users (Marines) and two Army domain experts which provides useful insight into the motivation behind the information needs of a dismounted group. This document was mainly used to gain insights into the goals and decisions in the goal hierachy. Furthermore the transcript of the workshop also provides a good example of an *"operational situation by a small group"*.

C - Cognitive workload during a Hasty Attack by the Royal Netherlands Marine Corps

This report provides a clear description of the way that recruits have been taught to act during a hasty attack and examines cognitive challenging tasks during these attacks [29]. Although this report is focused on the cognitive workload of the PC during a hasty attack, the detailed description of the hasty attack itself and the focus role that the PC plays in this attack was very useful and included the different goals that the PC pursues in different stages of the attack.

4.2.2 Tactical information

Two resources were used to obtain an understanding of the tactical and operational considerations of a PC. They were used to increase the understanding of the rationale of the PC during a hasty attack. Therefore these resources were mainly used to formulate the different goals of the PC and the decisions that need to be taken in order to achieve these goals.

D - Dutch Military Land Operations Doctrine Publication

The doctrine places the mission of the PC in a bigger perspective and indicates that every mission of a PC is always part of a larger mission. This document made clear what kind of information the PC receives before the start of a mission and why communication with the Squadron Commander (SQC) is important throughout the execution of the mission. The doctrine helped in distinguishing between responsibilities. The SQC commander commands multiple platoons and therefore he is also responsible for the relative positions of these platoons. The PC is responsible for the different sections in a single platoon. This also means that the SQC has to prevent fratricide between platoons where the PC has to prevent fratricide between sections. The document was mainly used to gain insights in the goals and decisions in the goal hierarchy.

E - Military tactics video of Max Richards

Max Richards served with British Special Operation Forces and is now focused on providing classes in military tactics. The tactical videos of Max Richards were specifically dedicated to the scenario of a hasty attack [30]. They provided insight into the different considerations that play a role for the PC in defining the attack plan, and it indicated important tactical skills like preventing fratricide. These videos also introduced a lot of military concepts which helped in learning to speak the military language, which benefited the interview with the former PC [7]. Because the videos focus on the considerations and possible outcomes, they helped in formulating both the sub-goal, decisions and a part of the SA requirements.

4.2.3 Evaluating the preliminary goal structure with colleagues

Based on the indirect information sources a preliminary goal structure was formulated. This was done to already identify a part of the missing elements before the interview with a former PC. This way the interview could be used as optimal as possible.

F - Peer review of the preliminary goal structure

The peer review was conducted with Maurice van Beurden and Lindsey Van Rooijendijk, who wrote the report [29] about the cognitive workload of the PC during a hasty attack that was used in the previous step and therefore have second-hand experience with PCs and the situation of a hasty attack. Although they both are not SME's, their experience in performing research on PCs helped in identifying missing elements. During this evaluation it was decided to focus the review with the PC on only a part of the entire GDTA, namely the goals that were expected to yield to most useful information: *"don't get killed', "decide whether or not to attack the enemy", "Plan of attack", and "defeat the enemy"*.

4.2.4 Evaluating the preliminary goal structure with a former PC

To finally validate and evaluate the goal structure an interview was conducted with a former PC.

G - Interview with a former platoon commander

The interview started with a short introduction to the research project and a short introduction to the experiences of the PC. After this, the entire goal structure was discussed with the PC, one subgoal at a time. This helped to improve the GTDA since the PC identified some missing SA requirements and clarified the decisions that are present in the goal hierarchy. A summary of the considerations that a PC makes during a hasty attack which became apparent during this interview are listed in appendix D and helped to define a the decisions in the goal hierarchy. To finalize the interview the PC was asked for his opinion on which goals he thought to add the most value to the PC with respect to increasing SA, which were used in the next chapter.

4.2.5 Observing the simulations of PCs performing a hasty attack

As a final step, a series of recordings of PCs performing a hasty attack in a mixed reality environment were observed to place the comments of the PC in context and check the goal structure a last time.

H - Watching simulations of platoon commanders performing a hasty attack

The recordings were made during an earlier experiment at TNO to test a simplified version of the Raptor with the PCs. In the test scenario, a PC would march with three sections to a set of waypoints. Around halfway through the march the platoon would make contact with the enemy. The PC was leading the platoon and therefore had to decide if and how to attack the enemy. During the experiment, the PC and the SCs could consult their Raptor. Three different runs were performed in total. Two different locations were used across the three runs of approximately 45 minutes. The recordings verified the earlier made comments made by the experienced PC.

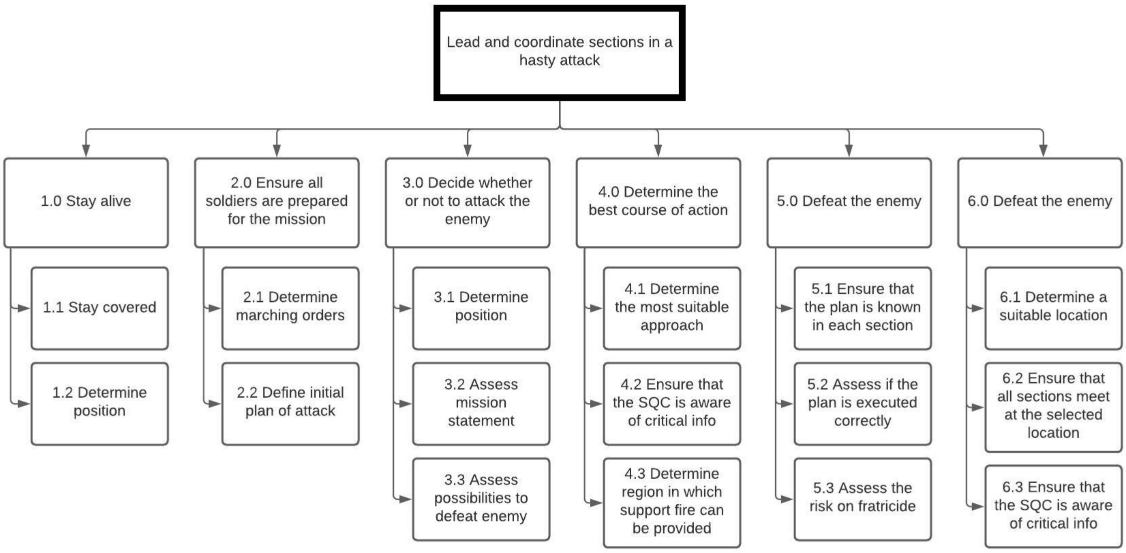


Figure 4.2: *Overview of the global goal structure*

Furthermore they also included radio communications between the PC, SQC and the SC's.

4.3 Results

The results of the GDTA is a detailed goal hierarchy. For this chapter, the identified SA requirements are the most relevant elements of this goal hierarchy. Image 4.1 lists all the unique SA requirements that were identified with the GDTA and groups them based on their SA level. Behind the requirements the different sources are listed in which these requirements were found or confirmed. Furthermore the goals and subgoals are also presented in image 4.2. A complete overview of the resulting goal hierarchy can be found in appendix C. This structure also includes the decisions that a PC needs to make to achieve his subgoals, and also shows which SA requirements are needed for a certain subgoal.

4.4 Discussion

Although the list of SA requirements became quite extensive, it is wise to assume that it is not complete yet. The main reason for this is that only a single interview with an SME could be conducted. Endsley argues that it can take somewhere between three and ten sessions to create an initial GDTA, after which she advises to use many more SMEs to validate the hierarchy [7]. In this project, that was simply not possible given the circumstances. Furthermore the subgoals *"start mission*

LEVEL 1	**Found or confirmed in source:**
Weapons of enemy	A, F, G
Location of the enemy	A, C, E, F G, H
Information that this location provides	G
Higher located ground	A, E, F, G, H
Mission objective	A, B, G, H
Rules Of Engangement (ROE)	D, G
Number of enemy troops	A, F, G, H
Weapons of enemy	A, F, G
Vehicles of enemy	A, F, G
Explosives of enemy	A, F, G
Skill level of enemy	A, F, G
Location of fire support section	E, F, G, H
Info from SQC	A, B, D, E, F, G, H
Update info from SQC	F, G, H
status of friendly troops	A, B, F, G, H
ammunition level of fire support section	A, B, E, F, G, H
approval to start execution of communicated plan	G, H
Location of DPC	A, E, F, G
Location of SC's	A, E, F, G
Location of the flanking section	A, E, F, G
Waypoints making up the plan	G, H
Region in which fire support is supposed to be provided	E, F, G
Safety angels for each section	B, E, F, G, H
Casualties	F, G, H

LEVEL 2	**Found or confirmed in source:**
Elements in terrain that provide cover	A, E, F, G, H
Range of enemy weapons	A, F, G
Expected information that an alternative location provides	G, H
Signs of possibilties for the enemy to receive backup	G
Techniques, tactics and procedures (TTP)	G
Ways to approach the enemy	A, E, F, G, H
Possible hidden enemy troops	G
Range of enemy weapons	A, G
Elements in terrain that allow for a covered approach	E, F, G, H
Possibilities to obtain optimal angel with point section (90°)	A, E, F, G, H
Liklihood of hidden enemy section	G
Areas to avoid because of risk on fratricide	B, E, F, G, H
Direction in which the flanking section is moving	E, F, G, H
Fire rate of fire support section	E, F, G, H

LEVEL 3	**Found or confirmed in source:**
Expected behaviour of the enemy	A, F, G
Future locations of friendly sections	A, F, G, H
Estimated time for flanking section to reach the enemy	B, F, G, H
Time that the fire support team can provide cover by fire	E, F, G, H
Moment at which the fire section needs to divert fire	B, E, F, G, H

Table 4.1: *Identified SA requirements*

with prepared soldiers" and *"set up secured position"* were not even discussed and validated with an SME, so it is especially likely that further improvements and completeness are possible for these goals. To finalize this statement, it can be assumed that one never knows if a GDTA is complete or not. To illustrate this, Endsley found four extra SA requirements by accident when she collected an objective assessment of the identified SA requirements and added four items as distracters to this list. They turned out be considered important after all, even with the huge amount of preparation and SME involvement that went into the creation of that GDTA [5].

It makes perfect sense that the more time is spent on refining the GDTA and the more people are involved, the more complete the overview will be. However, for this particular situation, I would argue that completeness is not the goal. The hierarchy needs to offer enough information to continue to the next phase, which means that higher-order SA requirements are identified that can be added to the Raptor in this case. Whether this is indeed the case is discussed in the next chapter, but it looks like this will be the case based on the number of identified higher-order SA requirements. At the same time, the process provided insight in the important consideration of a PC, which will be very useful in the design process later on in this project.

Furthermore, just four of the 43 identified SA requirements were identified during the interview with the PC. This indicates that the process of creating the GDTA based on indirect information sources was successful as well, although it was never suggested by Endsley. I would therefore argue, based on this anecdotal evidence, that in the case where a project wants to perform a GDTA and there is absolutely no involvement possible of the operator at hand, it will still be useful to perform the GDTA based on indirect information sources; the majority of the SA requirements can still be identified.

If future research wants to use a GDTA for a PC in a hasty attack, the goal hierarchy presented in appendix C will form a good starting point. I would like to advise these researchers to extend the hierarchy, when possible, by conducting extra interviews with SMEs, using the presented result as a starting point. Doing so will make the GDTA more complete and will further validate it. As mentioned, for this project the GDTA already provided enough options to choose from, but when more time will be spent on the GDTA I am sure that more SA requirements will surface.

The next step, based on the identified SA requirements, is to check which of these SA requirements are not yet present in the Raptor and can be turned into a functionality that can be included in a prototype within the time-span of the project. This follow-up step is performed in chapter 5.

4.5 conclusion

In conclusion, even though a slightly different approach towards the GDTA was followed, the desired result was nevertheless obtained. At the same time the process uncovered important considerations of the PC, which will be especially useful during the design part of this project. I would therefore suggest applying this way of performing the GDTA in other scenarios where SMEs are scarce as well.

The answer to the RQ1.1 is the collection of all the SA requirements that are listed in table 4.1. The PC uses all the components during a hasty attack. The information components are grouped based on their SA levels and can now be used to check which of them are available in the current version of the Raptor. By doing so it should become clear which elements are not yet included and therefore provide opportunities to improve the current version of the Raptor in how well it supports the PC in building SA. This comparison is made in the next chapter and should provide a selection of SA requirements that will be added to the Raptor.

Chapter 5

Selecting situation awareness requirements and combining them into new functionalities

5.1 Introduction

This chapter describes a selection process in which situation awareness SA elements are selected from the list that is the result of chapter 4. Including all the SA requirements that were identified in the previous chapter in the Raptor was impossible for practical reasons; the complexity would transcend the time-span of this research project and some requirements were already met by the Raptor. The research question answered in this chapter is RQ 1.2:*"Which important SA components are not already included in the current version of the Raptor and are technically realizable within the timespan of the re-*

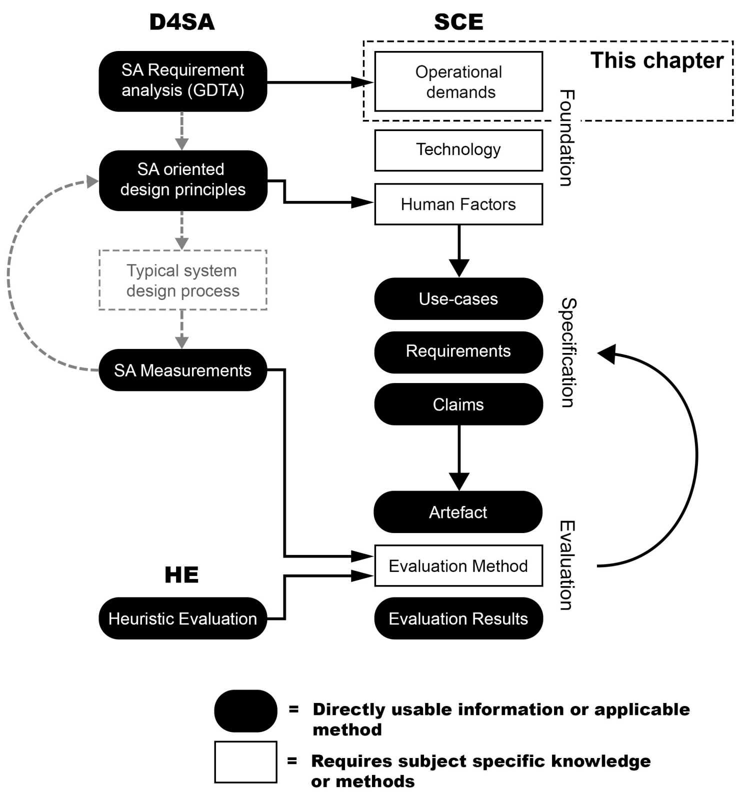


Figure 5.1: *Focus of this chapter with respect to the composite method*

search?

These SA elements were then used to define the problem description and further define the operational demands of the composite method as presented in chapter 3. The goal of this chapter was to end up with a set of SA requirements that will benefit the Platoon Commander (PC) and can be realised within the time-span of this project. In order to incorporate these SA requirements in the Raptor, several requirements were combined into functionalities. These functionalities then were be further specified, designed and developed and served as the problem description for the composite method.

We saw in the previous chapter that multiple SA requirements are used to make a decision in order to achieve a subgoal. By combining all the SA requirements that are needed for a single decision into a functionality, the PC can make this decision more easily and obtain the related subgoal more efficiently and effectively. This chapter elaborates on the process of starting with a list of SA requirements and ending up with a set of functionalities that can be designed and developed. The process in this chapter consisted of roughly three steps:

1. Compare the list of SA requirements with the Raptor to check which requirements are currently included.
2. Combine a group of SA requirements into a functionality which can be added to the Raptor and will benefit the PC
3. Evaluate these functionalities on their realisability within this project

A comparison was made between the identified SA requirements of the previous chapter and the current version of the Raptor. This way it became clear which SA requirements were already met by the Raptor and which were not. An SA requirement that was not met at all by the Raptor must be either (1) acquired visually, (2) acquired by communicating with a section or the Squadron Commander, or (3) estimated. Discrepancies between the information that the PC needs and the information that the Raptor provides also became clear by this comparison. These discrepancies could be decreased by implementing functions that add missing SA requirements to the Raptor.

The opinion of a PC was used to evaluate the added value of SA requirements that were not yet implemented in the Raptor. The PC gave his opinion about which sets of SA requirements would benefit the PC the most during a hasty attack. In doing so, the PC combined multiple SA requirements into functionalities that could be added to the Raptor. These combinations of SA requirements will provide the PC with all the information that he needs to make a decision that we saw in the goal structure of the previous chapter. In total the PC suggested three functionalities, which were all encompassing at least one higher-order level SA requirement.

The suggested functionalities are evaluated on two criteria to estimate how suitable they are for this research-project. These criteria consist of:

1. Does the technology that is required to create a functionality already exist?
2. Can a functionality be implemented or simulated within the time-span of this project?

In the next chapter the functionalities will be further specified and turned into a list of use-cases, requirements, claims, and a visual design

5.2 Method

Three different steps were taken to answer sub-question RQ 1.2. First of all, a comparison was made between the list of identified SA requirements and to what extent these were satisfied in the current version of the Raptor. Furthermore, user input from the PC was used to evaluate which combinations of SA requirements would add the most value for the PC. As a final step, the suggestions made by the PC were evaluated on how realistic inclusion in the prototype would be, be it simulated or fully functional. Each evaluation step is explained in more detail below.

5.2.1 Compare the list of SA requirements with the current Raptor

The goal of this comparison was to uncover discrepancies between the information that the PC needs and the information that the Raptor offers. This comparison was made in a table, consisting of the list of SA requirements and a color code that indicated to what extent the Raptor meets the SA requirement. This way it became clear in a visual way which elements were missing in the Raptor and therefore provided possibilities to improve it.

An SA requirement can be either (1) fully met, (2) partially-met (only some part of the information is provided, or it is not provided all the time), (3) it can be met by inferring from the system by a subject matter expert, or (4) not met at all. The resulting table can be seen in table 5.1.

5.2.2 Evaluate the added value for the PC of each SA requirement

The PC was asked what he thought would add the most value for a PC during a hasty attack. These answers consist of functionalities that can be added to the Raptor and that encompass multiple SA requirements. To further specify these functionalities the PC was asked to suggest functionalities that would include higher-order SA information.To make the PC benefit as much as possible, these functionalities were related to one of the decisions that a PC has to make. By including all the SA requirements that were needed to make this decision, it becomes very easy for the PC to make this decision and achieve the related subgoal. The suggestions of the PC were further evaluated in the rest of this chapter to see if they were suitable for the design and implementation phase.

This user input was the only direct user input in this research project in the entire creation process. That's why the opinions and suggestions of the PC weighted heavily in the selection and creation process of suitable functionalities to continue with. To understand the relevance of a functionality it was important to understand why the PC suggested this functionality. Therefore the PC was actively asked to specify his motivation after he suggested a functionality.

5.2.3 Realisability

The final step of selecting a set of functionalities was based on how suitable they were for this research-project. These criteria consisted of: (1) Did the technology that is required to create a functionality already exist? (2) Could a functionality be implemented or simulated within the time-span of this project?

The first part was evaluating the technology that is needed to meet an SA requirement. It was not the goal of this project to single handedly develop new technologies or algorithms. The focus was on implementing existing technologies. Therefore, a suggested functionality would only be selected for continuation in the next phase when there existed published research about this technology or there existed a comparable implementation, were it in another research area or not.

The second part of this evaluation was the estimated time that it took to implement a functionality in a prototype. The final goal of this project was to test the envisioned functionalities using a mixed reality environment. In this environment a mission could be staged and army units could be controlled to simulate a ground operation. In order to test the effects of a functionality with a PC in this environment, it should be possible to incorporate this functionality into the prototype and make it respond to the situation in this mixed reality environment.

As a bypass to both criteria one could argue that it was possible to simulate the use of a technology during the experiment. This would make it both possible to simulate the functioning of a yet not existing technology, and it could be used to simulate the use of a technology that would take a lot of time to implement. However, not all technologies were suitable for implementation by simulation during the experiment.

First of all, technologies of which no research had been published yet were not considered for implementation by simulation. Using technologies that don't exist yet was expected to raise a lot of questions with the test subjects. These questions would relate to how such a functionality would work and whether that is realistic or not. This would all divert from the question that is at the center of the planned experiment: what is the effect of the chosen functionalities on the SA of the PC. To prevent these distractions during the experiment, technologies that were non existing yet were not chosen for implementation.

Simulating a technology might be a solution for including a technology that is very complex when it should function on its own, but can easily be simulated by adding a human operator. An example of this are algorithms that understand language. Building a good solution for this can be complex, but simulating this is fairly easy: just let a human handle the meaning of the language and manually pass it into the prototype. This way the focus of the experiment will still be the effect of such an implementation on the SA of the PC, but the implementation time for the prototype is reduced a lot.

5.3 Results

5.3.1 Compare the list of SA requirements with the Raptor

The result of this comparison is visualized in table 5.1. The SA requirements that are already fully included are discarded in the rest of this chapter since they cannot be implemented in the Raptor anymore: they already are.

When the table is examined it can be seen that the only SA requirements that are fully met consist of level 1 SA requirements. Furthermore, we see that there are a lot of level 2 and level 3 requirements that are currently only met after inferring by

LEVEL 1	**Found or confirmed in source:**	
Weapons of enemy	A, F, G	
Location of the enemy	A, C, E, F G, H	
Information that this location provides	G	
Higher located ground	A, E, F, G, H	
Mission objective	A, B, G, H	
Rules Of Engangement (ROE)	D, G	
Number of enemy troops	A, F, G, H	
Weapons of enemy	A, F, G	
Vehicles of enemy	A, F, G	
Explosives of enemy	A, F, G	
Skill level of enemy	A, F, G	
Location of fire support section	E, F, G, H	
Info from SQC	A, B, D, E, F, G, H	
Update info from SQC	F, G, H	
status of friendly troops	A, B, F, G, H	
ammunition level of fire support section	A, B, E, F, G, H	
approval to start execution of communicated plan	G, H	
Location of DPC	A, E, F, G	
Location of SC's	A, E, F, G	
Location of the flanking section	A, E, F, G	
Waypoints making up the plan	G, H	
Region in which fire support is supposed to be provided	E, F, G	
Safety angels for each section	B, E, F, G, H	
Casualties	F, G, H	

= Fully met

= Partially met

= met by inferring by SME

= Not met by the Raptor at all

LEVEL 2	**Found or confirmed in source:**	**Present in the current version of the Raptor**
Elements in terrain that provide cover	A, E, F, G, H	
Range of enemy weapons	A, F, G	
Expected information that an alternative location provides	G, H	
Signs of possibilties for the enemy to receive backup	G	
Techniques, tactics and procedures (TTP)	G	
Ways to approach the enemy	A, E, F, G, H	
Possible hidden enemy troops	G	
Range of enemy weapons	A, G	
Elements in terrain that allow for a covered approach	E, F, G, H	
Possibilities to obtain optimal angel with point section (90°)	A, E, F, G, H	
Liklihood of hidden enemy section	G	
Areas to avoid because of risk on fratricide	B, E, F, G, H	
Direction in which the flanking section is moving	E, F, G, H	
Fire rate of fire support section	E, F, G, H	

LEVEL 3	**Found or confirmed in source:**	
Expected behaviour of the enemy	A, F, G	
Future locations of friendly sections	A, F, G, H	
Estimated time for flanking section to reach the enemy	B, F, G, H	
Time that the fire support team can provide cover by fire	E, F, G, H	
Moment at which the fire section needs to divert fire	B, E, F, G, H	

Table 5.1: *The extend to which an SA requirement is present in the current Raptor*

the PC. This inferring is a mental process that either requires the PC to look at level 1 SA requirements over time or base them on experience and gut-feeling. Since this inferring is a mental process, they will increase the mental workload of the PC a lot. A high mental workload could lead to the PC missing important information and events on the battlefield [1]

The partially-met requirements have different reasons why they are not fully met. One example is the group of requirements that relate to more detailed information about the enemy, like its vehicles or whether they have explosives at their disposal. The sections can communicate this kind of information using a text message in the Raptor, but they will not always do this. Previous research even showed that these text messages are not the preferred way of communicating these kinds of information [31]. At the same time the requirement that concerns the location of the enemy is also partly met, since the Raptor is not displaying the actual location, but shows the location that has last been communicated by a section.

Another partially-met requirement is *"signs of possibilities for the enemy to receive backup"*. In this case, not all the information is available. The PC can check whether there are roads and enemy military bases close to the location of the enemy, which would help enemy back-up sections to arrive at their location quickly. However, the Raptor does not show where other sections of the enemy are located, or whether the enemy possesses the possibility of air support.

Another partially-met requirement is *"the direction in which the flanking section is moving"*. This can only be determined over time, as the Raptor only shows the current location of each section to the PC, but this takes time. This information can be deducted from the waypoints that this section has passed by, but this does not reflect the actual movement direction of that section.

SA REQUIREMENTS	FUNCTIONALITY	TECHNOLOGY	PROTOTYPING DIFFICULTY
Location of the enemy (level 1) *Location of the flanking section (level 1)* *Direction in which the section is moving (level 2)*	*Monitor the execution of the communicated plan*	++ Technology exists	++ Easy to implement
Location of friendly sections (level 1) *Safety angles (level 1)* *Future locations of friendly sections (level 3)* *The moment at which the fire support team needs to divert fire (level 3)*	*Check the acute risk of fratricide and predict the future risk of fratricide*	++ Technology exists	+ Implementation can be complex but seems suitable to be simulated
Location of the enemy (level 1) *Likelihood of hidden enemy sections (level 2)* *Expected behaviour of the enemy (level 3)*	*Predict the movement of the enemy and predict hidden enemy elements*	- Technology exists on a different scale in different research area	- - Implementation is verry complex and not suitable for simulation

Table 5.2: *Summary of the realisability evaluation*

5.3.2 Using the input of a PC

The input of a PC resulted in three major suggested functionalities that would help the PC in achieving a subgoal in the goal structure of chapter two. The three functionalities are described below and listed with the SA requirements that are met by implementing such functionality. None of these functionalities are present in the current version of the Raptor and they encompass five higher-order SA requirements that are currently not met by the Raptor.

1. **Checking if sections move according to the plan**
   During hasty attack, the most important task of the PC is to monitor the execution of the plan. Part of this is checking how the sections are moving. Difficult terrain is often chosen for a flanking movement, as this allows for a hidden approach. At the same time, this makes it difficult for the flanking section to navigate. This means that a section can get lost or move too far, ending up behind the location from which it should attack the enemy. It is easy for the

PC to miss these events from happening as he is occupied with other tasks at the same time. Furthermore, these mistakes happen very gradually. The PC therefore mentioned that a PC would benefit a lot from a Raptor that would help him in monitoring if sections divert from the communicated plan. Sections moving away from their route have implications for the execution of the rest of the attack. When a section diverts from its route, it means that it will take this section longer to reach the enemy. This implicates that the fire support section should also be able to provide suppressive fire for this longer period of time.

Included SA requirements: location of the enemy (level 1), location of the flanking section (level 1), direction in which the flanking section is moving (level 2).

2. **Checking the risk for acute and predict the risk of future friendly fire**
Preventing fratricide was mentioned to be considered as one of the most important responsibilities of a PC. Friendly fire should be prevented at all times and a risk on fratricide is taken very seriously. The PC is solely focussed on the risk of fratricide between larger army segments like sections and platoons. Determining the risk of fratricide is currently deducted by the PC. At the same time the PC has to monitor the execution of the attack and communicate with the Squadron Commander. A Raptor that would help the PC to monitor these risks more easily and predicting future risks of fratricide would benefit the PC a lot, as it enables him to anticipate the situation.

   In a future risk of fratricide, it is especially important for the PC to know when a section has to divert its fire and in which direction. Getting help in this process will become more important in the future, as the army is using swarming tactics more often. This tactic is based on using more and smaller units, making it more difficult for commanders to monitor the risks of fratricide.

   Included SA requirements: Location of friendly sections (level 1), safety angles for each section (level 1), future locations of friendly sections (level 3), the moment at which the fire support section needs to divert fire (level 3).

3. **Always be aware of the location of the enemy and predict possible hidden sections**
The Raptor displays a static location of the enemy, which has been tagged by the section that made contact. For a good execution of the attack, it is important that everybody knows where the enemy is located at all times. In a real situation, the enemy is likely to move, unlike its representation on the Raptor which is static. The enemy will move to take cover or obtain a better firing

position. Being able to accurately predict the movement of the enemy would allow the Raptor to present the PC with the most likely current location of the enemy instead of the location where it was tagged most recently. Furthermore, it could help the PC if the Raptor would make an estimation were not yet encountered enemy elements are located. Right now the PC tries to predict this movement and possible hidden section based on experience, prior information about the terrain and information about how the enemy is organized.

Included SA requirements: location of the enemy (level 1), the likelihood of hidden enemy section (level 2), expected behavior of the enemy (level 3)

5.3.3 Realisability

The three different functionalities that were suggested by the PC are evaluated on their realisability. The result of this evaluation is presented for each suggested functionality individually. A summary is given in the table below.

Checking and predicting a risk on fratricide

The acute risk of fratricide can be checked quite easily. For each section a weapon danger area (WDA) can be constructed based on their current weapon, position and orientation [32]. The location of each section is already included in the Raptor and obtained by the blue force tracker. The only difficulty in creating a functional prototype is the fact that the orientation of a section is currently not present in the blue force tracker. Determining this orientation should not be technically difficult, as even current mobile phones can do so. In the mixed reality test environment, this information is readily available and therefore this information can perfectly be simulated.

Constructing the WDA can be a logical operation and is not expected to be a very time-consuming operation. In a real-life and urban scenario, there will be challenges in taking buildings into account when determining the risk of fratricide. However, there already exist open source repositories that are able to detect the presence of buildings on maps, so theoretically this should also be possible. Simulating this feature is even easier, as a ray-tracing technique can be used in the Unity rendering engine of the mixed reality environment.

The same ray-tracing technique can be used to predict a future risk of fratricide by tracing the rays from a point forward in time, based on the direction and speeds of each section. There lies a bit of work in realizing both functionalities, but due to the amount of already existing technology, like the possibility of ray tracing, it is expected that these functionalities are realizable within the time-span of this project.

Monitoring the execution of the plan

This functionality can be realized by using the current elements of the Raptor. The waypoints that the PC placed could be used to create a route for a section. The only modification that needs to be made is that the PC should give every section his own set of waypoints, which is expected to be perfectly possible, at least in a test scenario. The distance between a section and his route can then be used as a measure to check if he is diverting from this route or not. This means that this functionality does not need to be simulated at all.

Predicting the movement of the enemy and the possibility of hidden elements

This functionality consists of two parts. First of all, there is the task of predicting the enemy's behavior which can be used to present its most likely location in between the moments that this location gets tagged by a section. Predicting movement on its own should be possible and research has been performed that used machine learning techniques to train networks that are able to predict human movement, albeit on a different scale [33]. Implementing and training such a network on our specific case will be a time-consuming task.

Predicting the most likely location of not yet encountered enemy elements is a lot more difficult. Enemy elements come in a wide variety ranging from static elements like improvised explosive devices (IEDs) to moving sections and rapid moving vehicles. Therefore this functionality is not straightforward or comparable to other research or available solutions. This means that it is necessary to determine the best way to create a functionality like this in the first place, making the implementation of this a lot more time-consuming. A functionality like this could be simulated. In that case a human operator would just *"predict"* where all kinds of hidden enemy elements are located and the software would draw these locations on the screen of the Raptor.

Related projects to predicting these wide varieties of enemy elements have not been encountered. This causes such a simulation to draw a lot of attention to the reason why a technique like that would be possible. This diverts the attention of the PC from the effects that such an implementation would have on the SA of the PC

However, at another project within TNO an interesting result has been produced. This research uses supervised deep learning to come up with the most likely location of IEDs that are placed by an enemy that tries to stop a blue section. These projections are made based on the city map and the location of buildings. The result of this research is a technological framework, which seems not too difficult to implement, and will be suitable for testing. It is, however, much more narrowed down than the originally suggested functionality.

5.4 Discussion

All suggested functionalities seem to add value for a PC during a hasty attack. Also, they are not present in the current version of the Raptor and will make the Raptor meet extra SA requirements when they are implemented. More important is that these functionalities will add higher-order SA requirements to the Raptor, which was the goal of the project. The PC himself also suggested the functionalities based on the value they would add. Therefore it is believed that a PC can really benefit from these functionalities when they are implemented in a user friendly way.

The realisability evaluation is positive for two of the three functionalities. Monitoring the execution of the communicated plan of all sections is estimated to be very well realisable. Detecting acute- and predicting future risks of fratricide is considered to be more difficult, but here parts of the needed technology can be simulated very well. Therefore this functionality is also considered possible within this project. Realizing the functionality that would predict the movement of the enemy and the possible location of the enemy was considered to be not realisable within this project.

However, instead of discarding this functionality totally, adjusting the functionality makes it possible to include it within this research project. As mentioned, colleagues at TNO have been developing a technology that is able to predict the most likely locations of IEDs. This framework therefore already exists and might be included in this project as well. If the final technology seems to be too complex to implement, it is always possible to simulate the prediction of IEDs and just draw them to the screen on a location that is estimated by a human operator. The prediction of enemy elements is therefore changed to a very specific enemy element: IEDs. Now this functionality also seems realisable, it is also selected for continuing in the rest of the project.

The goal of this research project was to explore which role artificial intelligence (AI) can play in supporting the PC in his decision making during a hasty attack. Therefore in this discussion, some attention is paid to the way in which AI is involved in the suggested interactions. The main focus of this project is implementing AI that will provide the PC with higher-order SA information: information that is interpreted or projected.

All functions have in common that they take some information as input, process the input and act on it. These actions are triggering an alert or drawing predicted enemy elements on the map. Because of this sequence of input, processing, and action, we can consider all the suggested functionalities as AI. The way in which the processing is done differs between a rule-based system and a machine-learning based solution. The rule-based systems are used for the more simple computations and because of their nature, their output appears as logical to the PC. The machine learning algorithms are by their nature producing results can be harder to understand and trust by the PC.

Due to the situation of the hasty attack, it is very important that the AI parts provide the PC with information on which the PC can immediately take action. When the suggested functionalities are evaluated in this light then we see that they all do provide the PC with this information. After a warning computed by the monitoring AI, the PC can take action to change the course of a section or command a ceasefire to prevent fratricide. When making a plan the PC can immediately use the predicted locations of the IEDs by the predicting AI to come up with a plan that incorporates the risk of running into these. Even though the majority of the functionalities can suffice with a rule based AI, which is less complex than might be expected at the start of this project, it does provide the right kind of information that the PC needs.

5.5 Conclusion

This chapter started with an extensive list of all SA requirements that are used by the PC during a hasty attack. In three steps this list of SA requirements has been used to roughly specify three sets of SA requirements that can be included in the Raptor in the form of a new functionality. The steps that were take consist of:

1. A comparison between the list with SA requirements and the Raptor to check which requirements are currently met.
2. Combining a group of SA requirements into a functionality which can be added to the Raptor and will benefit the PC using user input.
3. Evaluating these functionalities on their realisability within this project.

Three functionalities are defined as a result. They are expected to add value for the PC, let the Raptor meet higher-order SA information, and seems reasonable to be implemented within the time-span of this project. In the next phase of the project, these functionalities will be designed and developed. The following functionalities will be used for that:

1. **Monitoring the execution of the communicated plan of attack** The Raptor will determine if a section is moving accordingly to the communicated path. It will use the distance between the location and its communicated route to do so. When a section moves away too far, the Raptor alerts the PC about it. The PC can then give new commands concerning their movement.

2. **Checking for acute- and predicting future risks of fratricide** Based on the location and orientation of each section the Raptor will determine if there is an acute risk on fratricide. At the same time the Raptor will predict future risks of fratricide between sections based on the current situation. The Raptor will alert the PC when a risk on fratricide occurs. The PC can then take action by either letting a section divert fire or command for a ceasefire.

3. **Predicting the location of IEDs** Based on the research performed elsewhere at TNO the Raptor will predict possible locations of IEDs based on the map of the current environment. The PC can use this information to take the risk of running into IEDs into account when making a plan of attack. Furthermore, the PC can warn sections to proceed with caution when they approach a suspected location of an IED.

Chapter 6

Specification and design of the new functionalities

6.1 Introduction

In this chapter, the functionalities that were selected in chapter 4 are specified in more detail and turned into a visual- and functional design. This step was important since it shows how such functionalities could be presented to the Platoon Commander (PC) and how the PC would interact with these functionalities. The challenge was to create a usable interface for functionalities that use artificial intelligence (AI) technologies. The process of creating this interface and the design choices that have been made for the final design are discussed in detail. In this chapter an initial design was created, evaluated and improved.

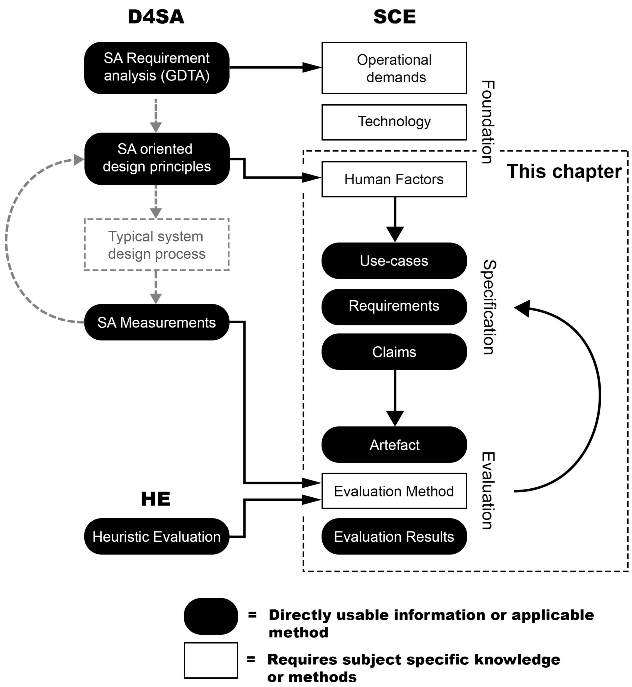


Figure 6.1: *Focus of this chapter with respect to the composite method*

To quote Endsley: *"Very often people must work uphill, against systems and technologies that block rather than enhance their ability to ascertain the information they need. Knowledge in vacuum is meaningless. Its use in overcoming human problems and achieving human goals requires the successful application of that knowledge in ways that are contextually appropriate"* [7]. Exactly this problem is why this chapter took the time to further specify and design the functionalities before they were being implemented.

This step in the entire process related to RQ1.3: *"How can the computed output be presented to the PC in an effective manner?"*. Effective here means that the PC should be able to directly understand the presented information and be able to take actions based on this. In order to find the answer to RQ1.3 all three existing methodologies that make up the composite methodology were used. .

The process treated in this chapter consisted of three major steps, consisting of: (1) creating an initial design of the functionalities, (2) evaluating the usability of this initial design, and (3) creating an iterated design. The elements of the different design methodology that were used for these steps are the following:

1. Socio-Cognitive Engineering (SCE): The specification phase of this method provides a process to iteratively create a system design. In doing so it combines and addresses the interaction between human cognition, technology, and context [6].

2. Designing for Situation Awareness (D4SA): The situation awareness (SA) oriented design principles that are listed in this method provide SA specific human factor knowledge [7].

3. Heuristic Evaluation (HE): Is used to evaluate the usability of the initial design without an subject matter expert [24].

The result of this chapter were both a system- and visual design which already included improvements to identified usability issues that were present in the initial design. Whether the final design indeed provides an effective way to present the computed information to the PC, like RQ1.3 was searching for, became clear after the user-test in chapter 7. In order to be able to perform this user-test, the design was be turned into a prototype as will be presented in chapter 6 first.

6.2 Method

As the only chapter in this project, this chapter used all the three existing methods that were used to create the composite method as described in chapter 3. The SCE method (explained in section 3.1.1) was used in combination with the D4SA SA oriented design principles (explained in section 3.1.2) to iteratively create the system- and visual design. The HE (explained in section 3.1.3) was used to evaluate an initial design on its usability. This evaluation was then used to rapidly improve the design by using the combination of the SCE method and SA oriented design principles again.

6.2.1 Initial concept development

The initial concept was created by iteratively and simultaneously defining the system design (use-cases, requirements and claims) and visual design. Knowledge of the hasty attack and military tactics that was collected in the previous chapters was used a lot in the creation of realistic use-cases and to take all the possible alternative steps into account. The 50 SA oriented design principles were used to check both the system- and visual design on a regular basis throughout the design process, and violations of the principles in the design were altered. The concept was constantly refined and extended by checking and refining existing use-cases, requirements and claims, creating new ones, and modifying and extending the visual design.

The system design was created in the Confluence environment that is made as an aiding tool to work with the SCE method by TNO and the TU Delft [34]. The visual design was made using Photoshop and screenshots that were taken from the existing version of the Raptor, which is discussed in more detail in section 7.2.1). The duration of the initial concept development was predefined by a period of six weeks due to planning related reasons.

6.2.2 Evaluation

The initial concept was evaluated using the HE method. This section explains how this evaluation was executed.

Participants

Four colleagues of TNO were used as evaluators. All the used colleagues have a background in human factors or interface design and already knew the researcher before participating.

Materials

The initial design that is the result of the first specification phase (and will be presented in section 6.3.1) was the subject of this evaluation. Both visual examples and a textual explanation of the design, similar like presented in section 6.3.1, were used to present the initial design to the evaluators. During the evaluation a printed version of the initial design explanation was used together with a printed version of appendix B, listing and explaining the nine heuristics. The researcher used a notebook to write down the comments of the evaluators.

Measurements

During each session the evaluator commented on violation of the nine heuristics that the evaluator observed. These comments were collected by the researcher.

Procedure

The evaluations were conducted with each evaluator separately. At the beginning of the evaluation the researcher gave a brief explanation of the project after which the initial design was presented. After this introduction the first heuristic was explained, and the evaluator was asked to search for violations of this heuristic in the initial design. Once the evaluator couldn't find new violations for this heuristic anymore the next heuristic was explained and evaluated in the initial design, until all the heuristics were treated. The researcher made sure to give the evaluator as much time as needed to think and made sure to not provide suggestions. After the evaluation of all heuristics the evaluator was thanked for participating.

6.2.3 Creation of the improved concept

After the evaluation of the initial system design an iteration on this design was made to overcome some of the identified problems. First of all the identified issues were compared with the SA oriented design principles. In cases where the usability heuristics and design principles were contradicting, the SA oriented design principles were followed in the improved concept to keep the focus on supporting SA. Again, the SCE method of specifying and modifying the existing use-cases, requirements and claims, and using the SA oriented design principles were used again in this process.

6.3 Results

This section is divided into three main subsections. The first subsection discusses the design of the system that is the result of the SCE and D4SA methods. A complete list of use-cases, functional requirements and claims on which this design is based can be found in appendix E. The concepts behind the initial design are first explained and accompanied by images that visualize them. Once this initial design has been discussed, the second subsection treats the evaluation of this design and the most important results of the HE of this design are presented. The last subsection shows the iterated design in which a part of these issues is resolved.

6.3.1 Initial Design

The SCE method resulted in concept design that is based on a list of use-cases, requirements and claims that can be found in appendix E. A quick overview of the claims is presented in table 6.1. These claims serve as a testable hypothesis of the effects of the solution. The resulting system- and visual design consists of more than just a notification when the Raptor detects an alarming situation for one of the three functionalities that are implemented. Extra concepts have been added to the system to improve the interaction with the Raptor and aim to optimally support the SA of the PC. As discussed, the three functionalities monitor all the sections in different ways. Whenever one of these functionalities is triggered, i.e. either a section diverges from its route, a (future) risk of fratricide is detected or a section approaches an improvised explosive device (IED), this is referred to as a mission critical cue (MCC).

CL001	*The Raptor reassures that the PC will be aware of sections that diverge from their planned route*
CL002	*The Raptor reassures that the PC will be aware of sections that end up behind enemy lines*
CL003	*The Raptor reassures that the PC will be aware of occurrences of acute risks of friendly fire*
CL004	*The Raptor reassures that the PC will be aware of future risks of friendly fire*
CL005	*The Raptor reassures that te PC creates a plan that anticipates on the presence of IEDs*
CL006	*The Raptor reasures that the PC validates self made predictions of IED locations*
CL007	*The Raptor reassures that the PC does not receive redundant notifications*
CL008	*The Raptor reassures that the PC stays in control of which information he receives*
CL009	*The Raptor reassures that the PC will always able to interact with the Raptor.*
CL010	*The Raptor reassures that the PC will not miss a section approaching a predicted IED located.*
CL011	*The Raptor reassures that the PC will not make a plan of attack that will cause friendly fire with another platoon.*

Table 6.1: *Summary of the claims for the initial concept design*

Global SA

Image 6.2 shows some more general modifications that have been made to the design of the Raptor order to improve the global SA of the PC. The way in which a section is displayed on the map has been slightly modified, in accordance with the latest version of the NATO joint military symbology [35]. This way the exact location of a section becomes more clear, as well as the fact that the PC is able to see the direction of the section at a glance. At the same time the previous locations of the section are visualized by the grey dots, indicating how the section arrived at their current location.

Furthermore, an area on the map is shaded, indicating that the enemy should not be approached from within that area. This area represents the no-go area, as firing at the enemy from within this area might result in the occurrence of friendly fire with a friendly platoon that is not directly visible on the Raptor, but is located within firing range.

Finally, a dashed red line is drawn through the location of the enemy, representing the frontline. In their approach, sections should not end up behind this line. There could be undetected enemy sections there and it increases the risk of receiving friendly fire from the fire support section.

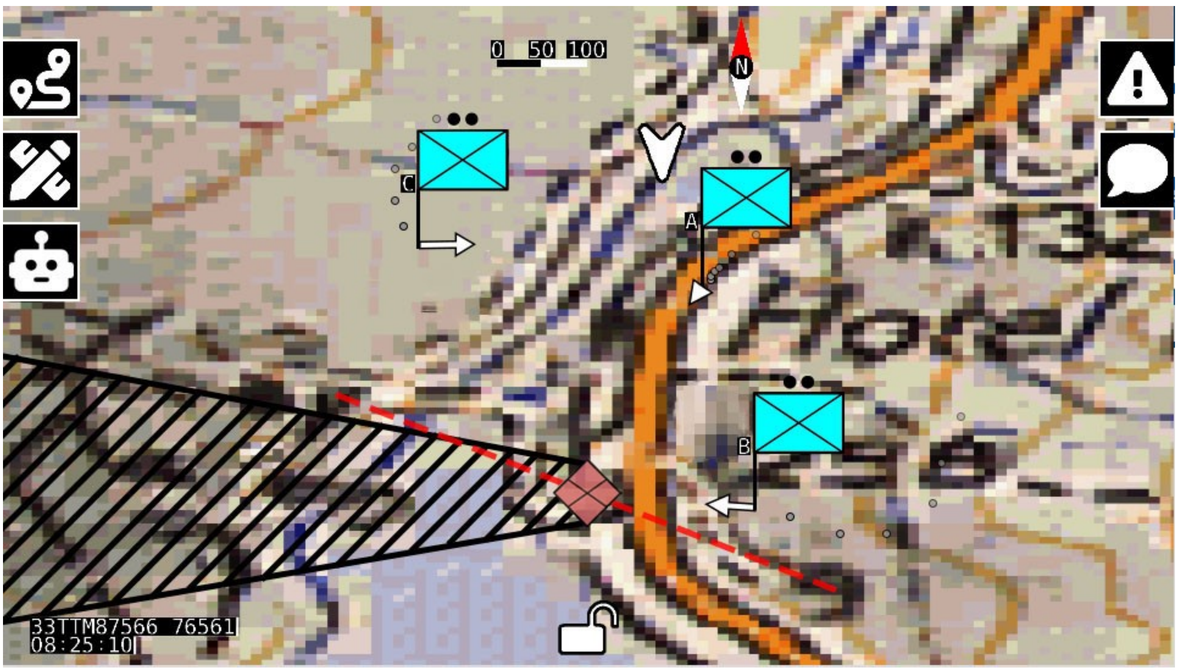


Figure 6.2: *The overview screen of the initial design*

The alarms

The main part of the design is the way in which the three functionalities defined in chapter 5 alert the PC when an MCC is detected. The goal of these alarms is to notify the PC about a situation that is important for the PC. In order to draw the attention of the PC the alerts need to be salient. Once the PC sees the alert, he needs to be able to assess the entire battlefield to the best of his abilities to determine which actions he will take based on the current situation. For this reason all the alerts have two different visual ways of representing them: a salient one to draw the attention of the PC and a more generic one to let the PC better assess the entire battlefield again. These two different ways of visually representing the alert to the PC are shown for each functionality in image 6.3, 6.4, and 6.6. The visualizations of the alerts are discussed for each functionality individually.

Diverting sections are outlined by a blinking red line to draw the attention of the PC. The visualization of the alert can be seen in image 6.3. The effect of the diversion is indicated to the PC by showing the time that it will take the section to go back to the route again. A dashed line represents the quickest way to return to the route. The green part of the line indicates the area within which the section is no longer considered diverging anymore. In the general visual representation of this alert the blinking red outline is removed, and the time indication is also turned from red into black.

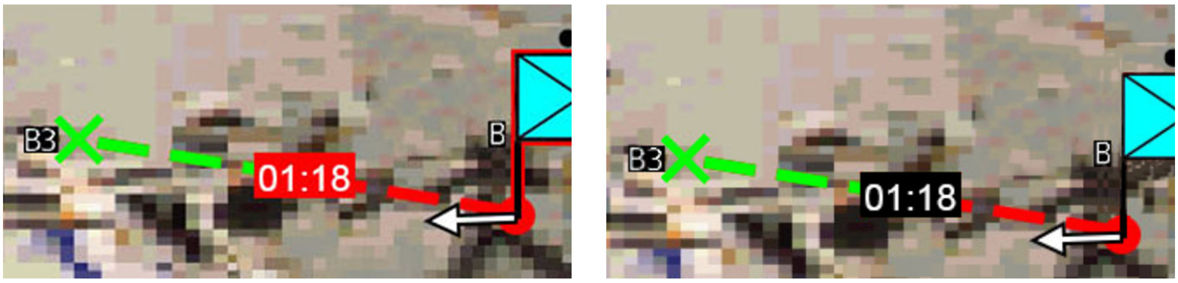


***Figure 6.3:** Difference between a salient (left) and general (right) visual alert for a section that is diverging from its route*

Predicted future risks of fratricide are presented to the PC by placing a red triangle with an exclamation mark on the location where the Raptor predicts the risk of fratricide to happen. This icon is chosen because the shape of a triangle and red color convey danger. The exclamation mark is chosen because the former PC indicated that fratricide is the most important risk and should be prevented at all times during the evaluation of the Goal-Directed Task Analysis (GDTA) goal hierarchy in chapter 4.

A time indication below the icon tells the PC how much time there is left before the Raptor expects the risk to become acute, as can be seen in image 6.8. A red blinking outline indicates the section that is taking the risk and a dotted line indicates the distance between this section and the location where the fratricide risk will become acute. The weapon danger area (WDA) of the section providing the fire that causes the risk of fratricide is indicated by blurring the map in that area, and the side of this WDA in which the section at risk is moving is highlighted by a red dashed line.

In order to prevent clutter, the Raptor is focusing on the future risk of fratricide that will occur the soonest. In image 6.4 a second, smaller fratricide icon is placed on the map as well, indicating that there are two sections that provide a risk, but at the same time allowing the PC to focus on the most urgent problem.

The general view of the alert just shows the predicted location of where the risk will become accurate. A dotted line still indicates to which section this risk relates, but the time indication is moved. It is now located next to the section, where it is positioned next to a small fratricide icon. This way the PC can easily see which section has a risk of future fratricide and how much time is left before the risk is expected to become acute.

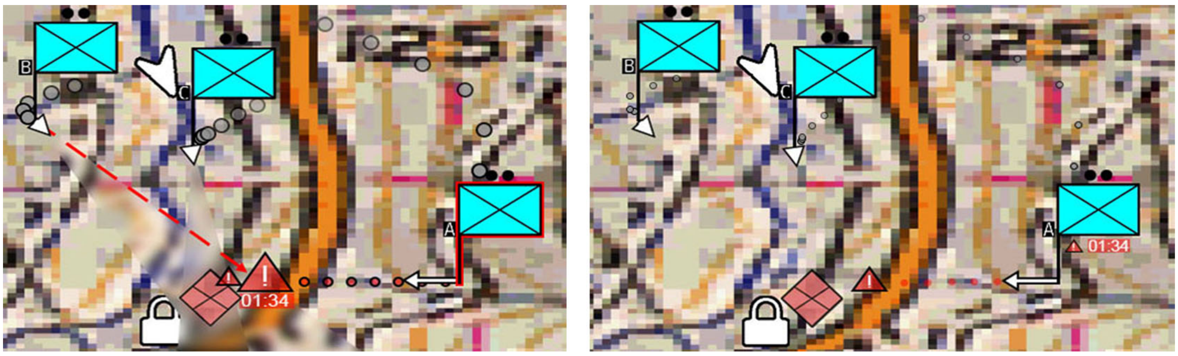


Figure 6.4: *Difference between salient- (left) and general (right) future fratricide visual notifications.*

Acute risks of fratricide will always be displayed in a salient way. This is done by placing the fratricide icon on the location of the section and by showing the WDA(s) of the section(s) that are providing the fire with a red fill as can be seen in image 6.5.

IED predictions are visualized like can be seen in image 6.6. The IED icon is in accordance with the latest version of the NATO joint military symbology [35], with the textual addition of the certainty of the prediction of this IED location. The certainty of the prediction is also represented by the opacity of the IED icon. Lower certainty

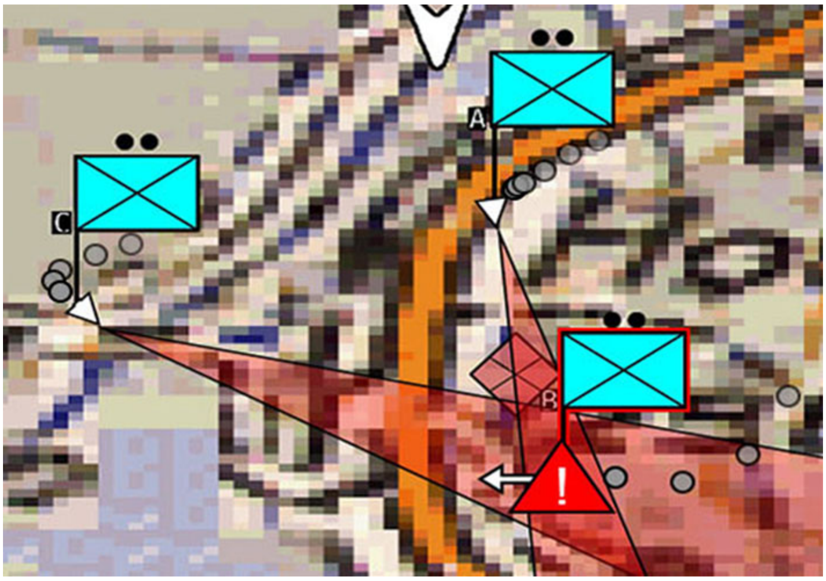


Figure 6.5: *Visualization of an acute risk of fratricide*

is represented by a higher opacity.

When a section gets within a threshold distance of an IED the salient alarm will show. This highlights and outlines the IED which a section is approaching and draws a line from this section to the IED. The section self is also highlighted by a blinked red outline. Finally, a time indication next to the IED shows the estimated time before the section reaches the location of the predicted IED.

The general alert that indicates that a section is approaching an IED just shows a dotted line between the section and the IED. The time indication of the estimated time when the section reaches the IED is moved below the section icon, accompanied by a small IED icon.

Auditory notifications

From the claims of the SCE method, it followed that the PC always needs to be notified when an MCC is detected. In practice, the PC will not always look at the screen of the Raptor, however. When the PC is moving the Raptor is often tucked away. It might happen that the PC is looking at his direct surroundings instead of at the Raptor for a long time. In order to make sure that the PC will always be notified about a detected MCC, the prototype will also be able to give auditory alerts.

The flowchart in image 6.7 indicates when which kind of alert is given. The auditory alerts are used when the PC is not looking at the Raptor. This can be the case when the Raptor is tucked away or when the PC has not looked at the screen

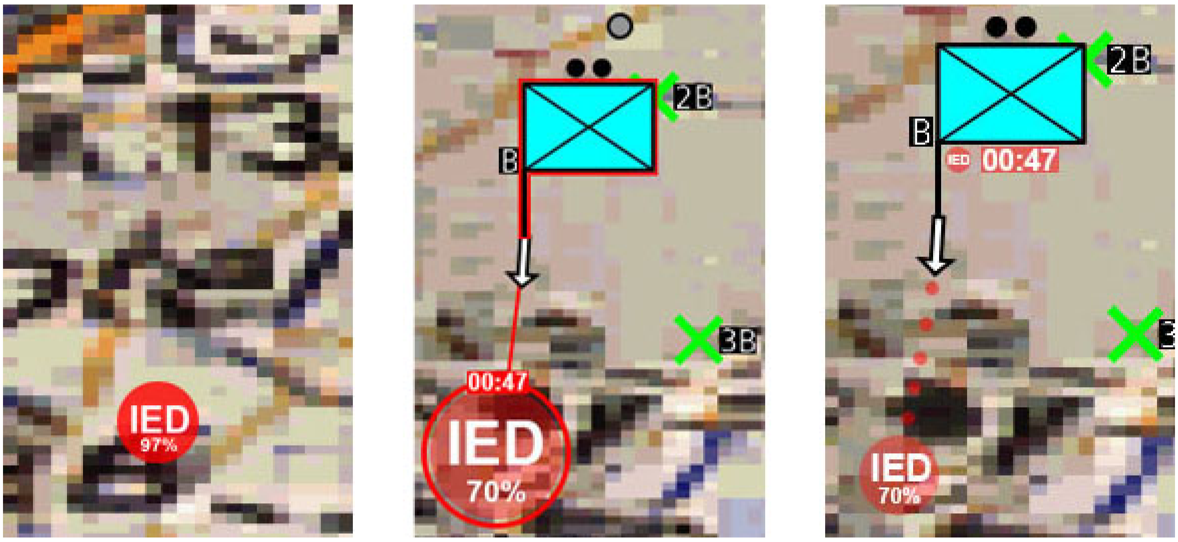


Figure 6.6: *Example of how IEDs are presented (left), and the salient- (middle) and general (right) notifications that are shown when a section approaches an IED.*

in the last 10 seconds. When the PC is looking at the screen, a salient visual alarm is given for 10 seconds, after which it turns into a general alert. If the PC was not looking at the Raptor but opens it after he received the auditory alarm, the Raptor will display the salient alert for 10 seconds from the moment that the PC looks at it and if the MCC is still present. The only exception to this is an MCC that concerns an acute risk of fratricide. In that case always an auditory- and a salient visual alert are given.

The PC has two options to view an alert in more detail. Using the menu, the PC can select a certain functionality that presents all the related MCCs, for example, all sections that are diverging from their route, in the salient modality. When a functionality is viewed in the salient mode its icon in the menu will be filled yellow, just like the bot icon in image 6.8. To view all the MCC's of a single section he can also tap on this section, and all the MCC's for this section will be displayed in the salient modality. The PC is always able to return to the general view of all alarms by tapping somewhere on the map or tapping the same button again. When he does so all alarms return to their current state and the menu closes completely.

Controlling the functionalities

The PC is able to (temporarily) turn a functionality on and off. The easiest way to do so is manually using a set of buttons. The SCE method showed that the PC will not always look at the screen of the Raptor and therefore the PC will also not always be able to turn off functionality manually. For this reason, the Raptor will also be able to understand voice commands, which the PC can use to turn off the functionalities as

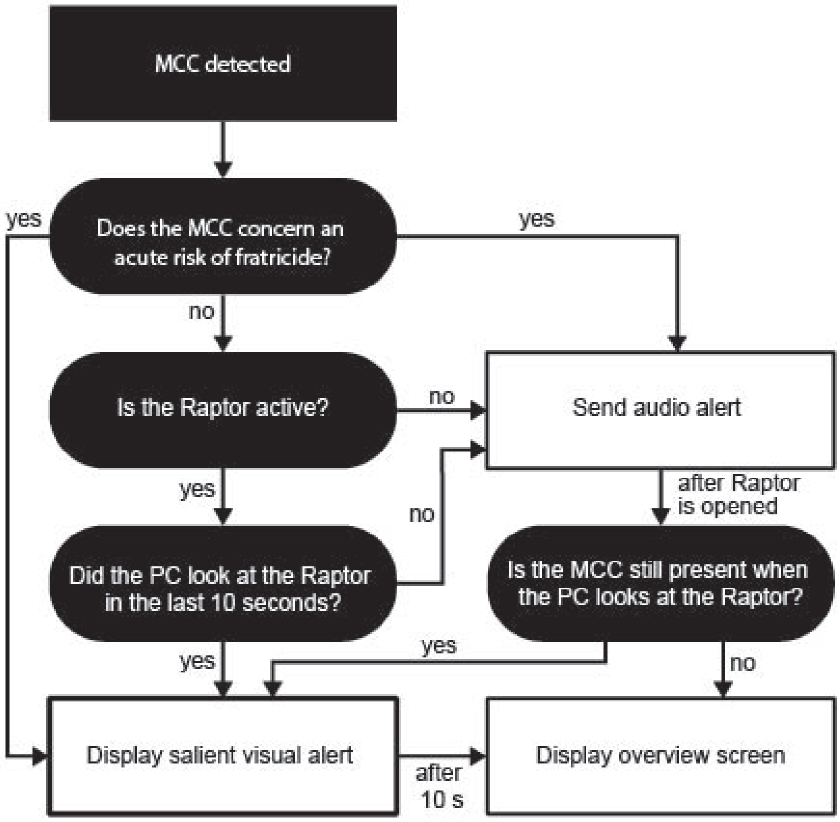


Figure 6.7: *flowchart that indicates when each modality of alerts is used.*

well.

By allowing the PC to turn off functionality the Raptor anticipates the fact that it might make a mistake. This way the PC can easily remove the related alarms from his display. At the same time, this also allows the PC to prevent alerts from showing up when he deliberately creates an MCC, which can happen in unforeseen situations.

In some cases, it might be necessary for a section to move away from its route due to the current state of its surroundings, or a section will actively approach a location on which an IED is expected because the situation simply requires this. Whatever the reason might be, whether it is a situation that triggers an MCC or a technical misinterpretation, the PC will always be able to stop the alerts, keeping him in control. Keeping the operator in control at all times and allow him to overrule the automation is also in line with the D4SA principle 37 *"keep the operator in control and in the loop".*

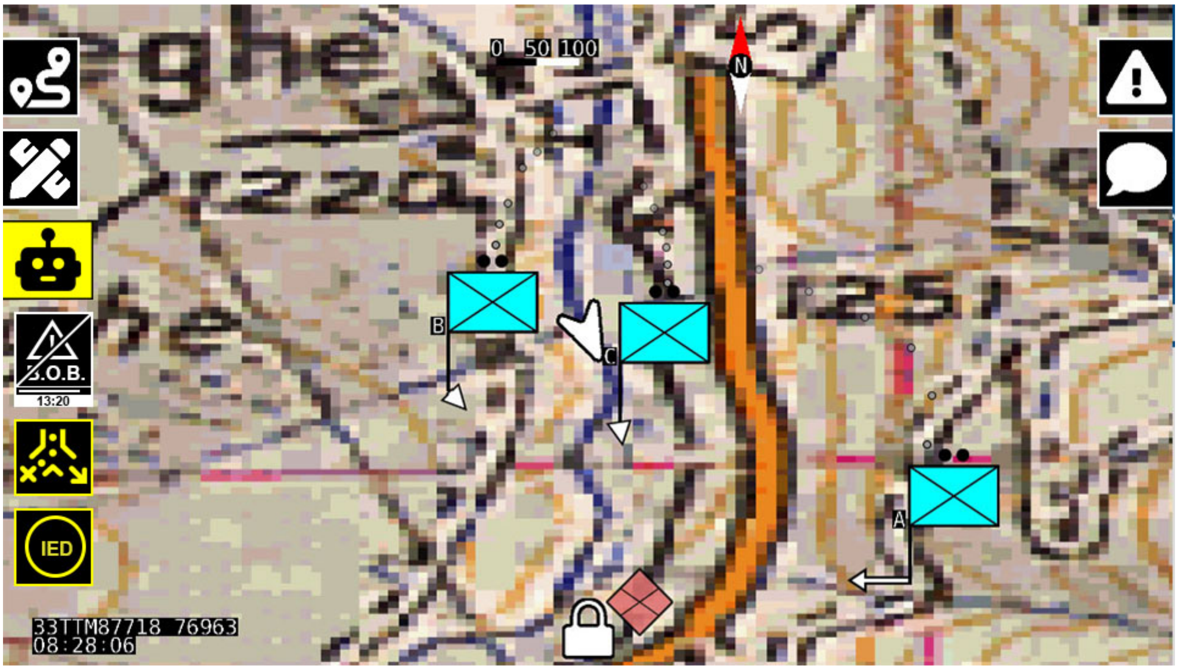


Figure 6.8: *The fratricide detecting and predicting functionality is turned off*

As a means to prevent the PC from forgetting to turn the functionality back on, the PC is also able to temporarily turn a functionality off. In that case, the monitoring functionality will automatically be turned on again after a specified time. Using a voice command, the PC can specify any duration he wants, where the manual interface presents him with a small set of predefined options. When a functionality is turned off, the icon of that functionality will be crossed out by a white line. A small clock and progress bar appear below the icon that indicates how long that functionality is still turned off.

6.3.2 Heuristic Evaluation

The HE resulted in a list of 35 usability issues. The complete list of issues and the table that indicated which evaluator found which issue can be found in appendix J. In this section, the most relevant findings of the HE are discussed per heuristic (numbered and displayed in bold below).

1. Simple and natural dialogue

Red is used a lot in the warning symbols (issue 1). The functionality in which this becomes the biggest issue is the risk of (future) fratricide. These alerts are often given near the location of the enemy. This leads to a lot of red shapes in the area close to the enemy, which can lead to clutter and the loss of salience of the alert. Furthermore, a lot of new buttons that were introduced were considered to be unintuitive (issue 3,4, and 7).

2. Speak the user's language

The HE did not detect any usability issues with respect to this heuristic

3. Minimize user memory load

One evaluator commented that the functionality that monitors sections diverging is not showing the correct information (issue 9). The Raptor should not indicate the time to the route, but the extra time that it takes to get to the last waypoint. This issue is in line with the D4SA principle 3: *"provide support for level 3 SA projections"*. The PC needs to know what the impact of the detour is on the arrival time at the enemy. In other words: how much longer does the fire support time need to be able to suppress the enemy.

Usability issue 12 indicates that the functionality that predicts a future risk of fratricide only clearly indicates the most urgent future risk and is neglected the other risks of future fratricide. This issue is in line with the D4SA principle 8: *"use information filtering carefully"*. By only showing the first expected risk on fratricide the design is trying to prevent a cluttered display. However, as explained in D4SA principle 8, this type of information filtering can actually degrade SA. By not equally visually indicating both the risks, the PC might miss the second risk of fratricide. The difference between being aware of both the risks or just the most urgent can make a difference for which commands the PC will give to the fire- and flanking section. Furthermore, the way the WDA is visualized in the alert of a future risk of fratricide is barely visible (issue 10).

Finally, it was mentioned that auditory alerts might increase the workload of the PC by putting extra messages on an already occupied audio channel (issue 14).

4. Be consistent

The fratricide icon is also used in the interface to tag an enemy location and therefore has two different meanings (issue 16). The same goes for the dashed red line which is used on the map as grid lines, in the future fratricide alert to indicate the side of the WDA where a risk of fratricide is predicted, and it is used to indicate the front line of the battlefield (issue 18 and 21).

At the same time, a lot of red is used in the visual alerts for all three functionalities, while red elements were previously only used for a tagged enemy (issue 24). Furthermore, a section that is outlined red also has two meanings (issue 27): it can either mean that this section is selected or that an alert is given for this section.

5. Provide feedback

Three evaluators noted that the Raptor does not reply to a voice command. Because of this, it is unclear whether the Raptor understood the PC and what action the Raptor is going to take (issue 28).

6. Provide clearly marked exits

The HE did not detect any usability issues with respect to this heuristic.

7. Provide shortcuts

The most important issue that was identified was the fact that there is no way to skip the 10 seconds of salient view when an alarm is given (issue 29). This identified issue is also in line with D4SA principle 37 *"keep the operator in control and in the loop"*. Furthermore, it is not possible to drag items like waypoints and enemies around to move them (issue 31). Finally, there is also no way to delete all waypoints for a section at once (issue 30). Each waypoint needs to be removed individually in the initial design.

8. Good error messages

In the initial design the Raptor does not give a single error message. The Raptor also does not give an error when it did not understand a voice command of the PC (issue 32).

9. Prevent errors

Highlighting of the selected section could in a future prototype provide clarity for the PC when placing waypoints (issue 33). Furthermore, the Raptor could also warn the PC when a waypoint is placed behind enemy lines (issue 35).

The majority of the usability issues were identified by only a single evaluator. This does not necessarily indicate anything about the importance of this issue, but more about how apparent this issue was. Nevertheless, it seems reasonable to at least look more closely at issues that are identified by multiple evaluators. The following issues were identified by at least two evaluators:

- 8: Due to the shape of the triangle, it is ambiguous which way a section is looking that is not moving.
- 10: The WDA in the visual alert for a future risk of fratricide is not salient enough.
- 16: The triangle with the exclamation mark is also used in the interface for tagging an enemy and therefore has two different meanings.
- 19: Important areas, like the no-go zone and the WDA's during a (future) risk of fratricide, are displayed in different ways.
- 28: The Raptor does not reply to a voice command of the commander. Therefore the commander does not know if the Raptor understood him and if the settings are changed or not.
- 29: When the Raptor presents the salient view for ten seconds, there is no way that the commander can return to the overview quicker.
- 31: It is not possible to move around objects like waypoints or an enemy location.
- 35: The Raptor does not warn the PC when he tries to place a waypoint behind the enemy line.

6.3.3 Iterated Design

In this section, it is presented how the results of the HE were processed. The majority of the identified usability issues (23/35) have been used to improve the design by solving these issues in an iterated design. The issues are discussed per the heuristic to which they belong. For each heuristic, the solutions to issues that are included in the initial design are presented first after which the reasons for not implementing the remainder of the identified issues are discussed.

An exception is made for issue 15, 18, 20, 21, 22, 23, and 35 as these relate to two functionalities that have been discarded in the iterated design. It concerns the visualization of a no-go area and highlighting an area in order to prevent fratricide with another platoon. The reason for discarding these functionalities is discussed after all the other identified usability issues are treated. The visual representation of the iterated can be seen in image 6.9, 6.10, and 6.10.

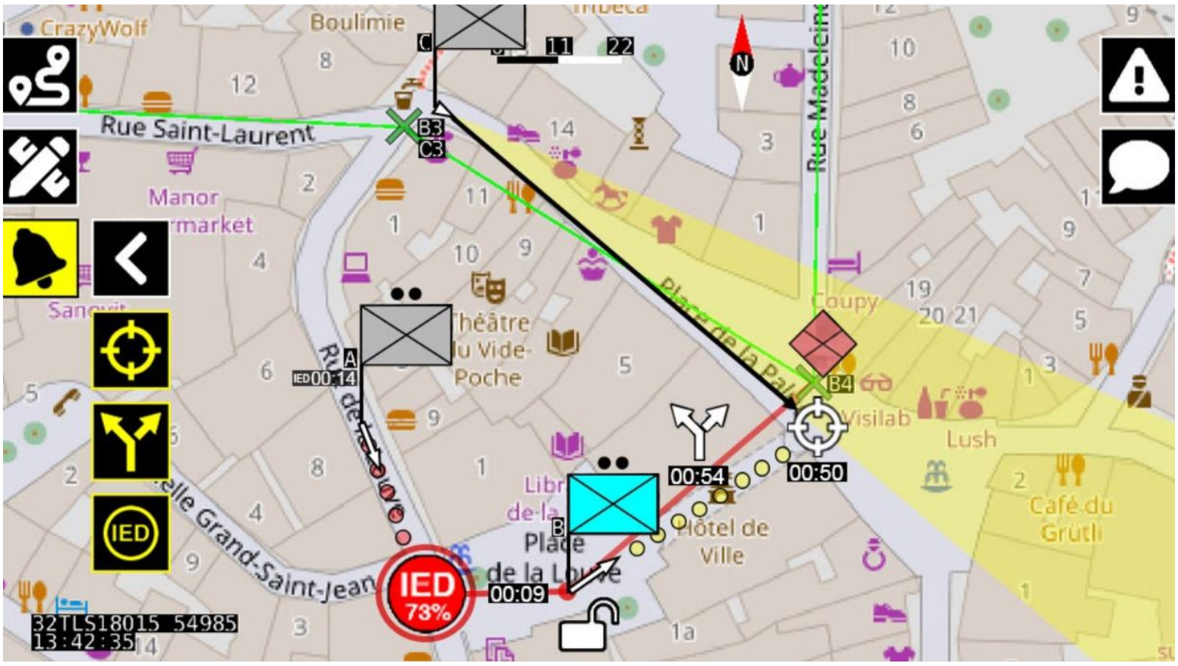


Figure 6.9: *The new buttons of the menu structure. Section B is selected to show all the salient versions of the visual alerts of all the functionalities at the same time.*

1. Simple and natural dialogue

The fratricide warning icon has been changed from red to white (issue 1). The IED icon is still displayed in red since this is a predicted enemy element, with which red elements are associated. The icon on the button to access the three alerting functionalities has been changed to an icon that better represents the subject of the sub-menu that will be opened: alerts (issue 4). The fratricide icon itself is also been changed (issues 3 and 7).

The unfolding structure of the menu (issue 2) is not solved in the iterated design since it would require a redesign of all the other menu structures in order to stay consistent (which is also one of the heuristics). Since time is limited and this is the

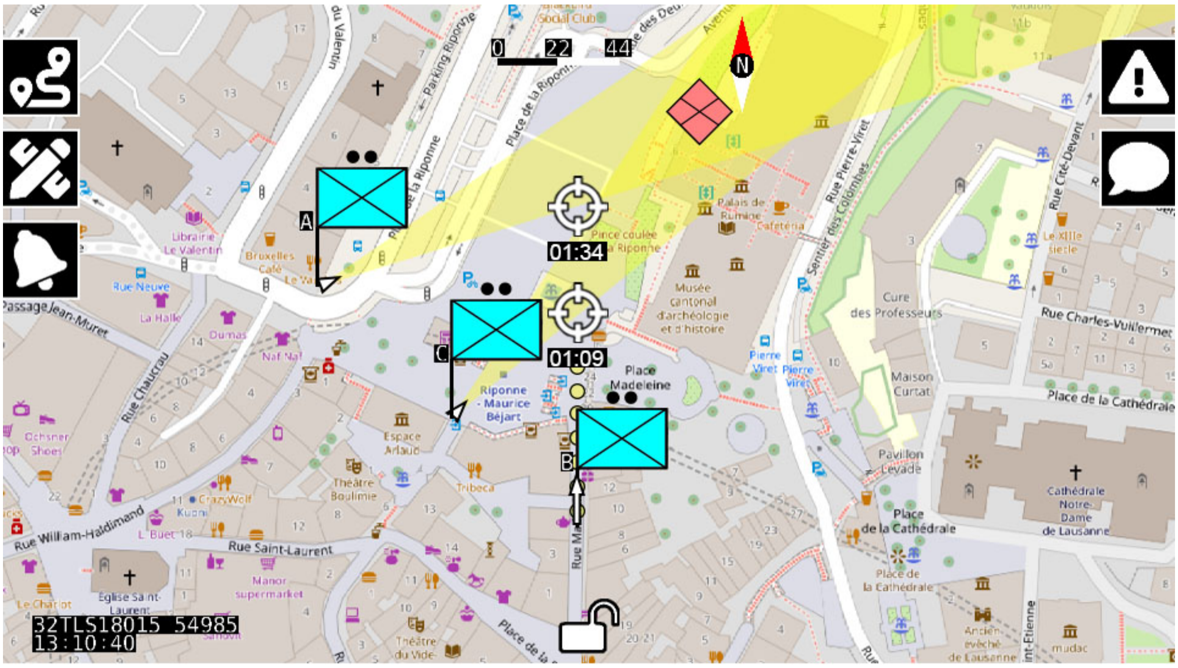


Figure 6.10: *Visualization of two consecutive risks of friendly fire in the iterated design.*

first design of the new functionalities this improvement step is neglected in this design iteration.

Holding a button is indeed a new interaction (issue 5). However, it is a great example of a shortcut that a more experienced user can use (which is in line with the heuristic of providing shortcuts). At the same time, it is an interaction that is often present in touch devices, which would make it more logical for the PC to also try this interaction on the Raptor. Finally, issue 31 suggests the possibility of dragging the icon of a waypoint or tagged enemy around in order to move its location. This interaction is also integrated into the iterated design and therefore holding down an icon in order to access a different interaction will not only be implemented in other places in the interface as well.

Finally, the auditory notifications in the iterated design will also still contain a lot of formal language (issue 6). This is a conscious design choice since it is in line with how military personal communicates and therefore adheres to the heuristic of speaking the user's language. The *"formal language"* in this issue can also be referring to the fact that the auditory messages are quite lengthy. This is also done on purpose to let this be in line with D4SA principle 27 *"Make alarms unambiguous"*. Presenting the auditory notification this way explains the risk in more detail (section diverging in eastern direction) and enables the PC to put the alert better into context. The identified usability issue suggests that the PC doesn't need this much informa-

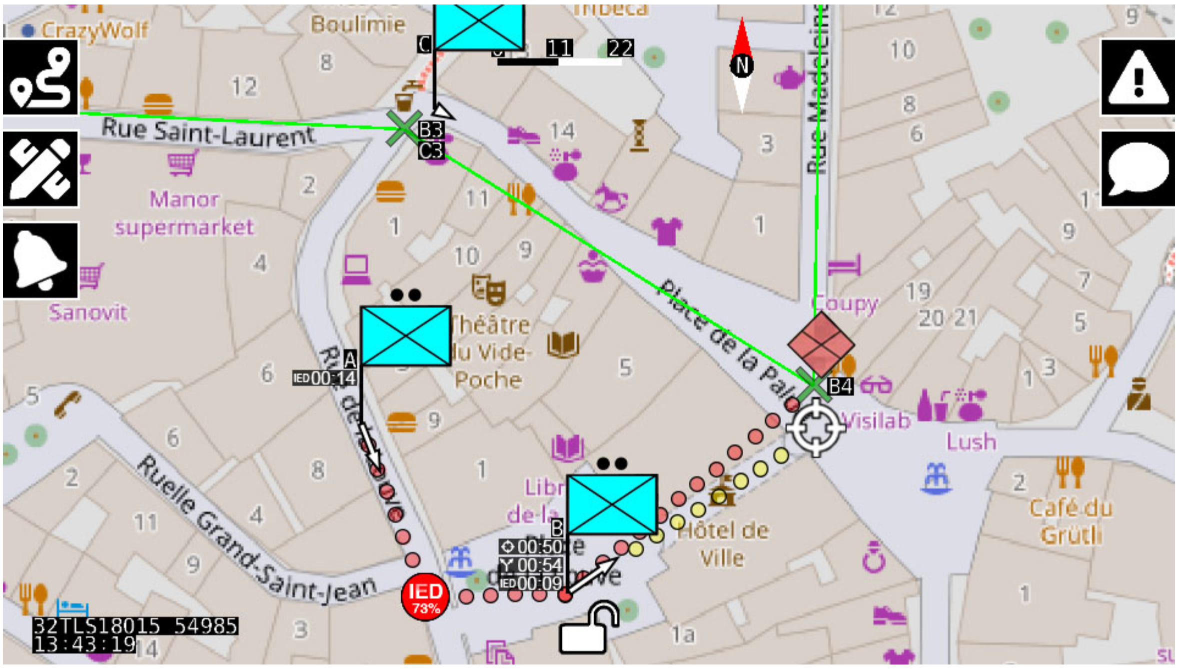


Figure 6.11: *The visualization of the general visual alerts of the iterated design.*

tion via an auditory alert and that a very brief alert could also suffice. Whether the PC will prefer a longer or a shorter auditory notification will turn out in the user-test.

3. Minimize the user's memory load

The triangle of the motion vector of each section has been made more pointy, making it less ambiguous in which direction it is pointing when a section is not moving (issue 8). At the same time, the WDA in the visual alert of a future risk of fratricide is visualized in yellow now instead of as a blurred background (issue 10). Furthermore, the progress bar that showed for relatively how long a function would still be turned off is also removed from the iterated design (issue 11).

The visual alert for a section that is diverging from its route has also been changed. The alert now indicated the closest route to the next waypoint and indicated the extra time that the section needs to arrive at their destination (issue 9). At the same time, the alert that indicates a future risk of fratricide now also shows consecutive risks of fratricide in a way that represents both the cases as equally important (issue 12) and by doing so is more in line with D4SA principle 8 *"use information filtering carefully"*

Image 6.11 shows a situation in which all three functionalities issue an alert at the same time and one section even has an alert from all the functionalities. And still, even though this situation represents a very chaotic situation, the designer thinks that the display did not get too cluttered to understand. Therefore issue 13 *"the display will be very cluttered when multiple alarms go off at the same time"* is neglected.

Finally, the iterated design will still contain auditory messages (issue 14). The reason for this is that these messages are considered by the designer as a major improvement because they allow the PC to be aware of an alarming situation at all times, also when the Raptor is tucked away. This interaction is also needed to meet the claims CL009 which states that the Raptor should reassure that the PC will be able to always interact with the Raptor, also when this is tucked away.

4. Be consistent

Issues 15, 18, 20, 21, 22, and 23 are discussed at the end of this subsection as it relates to a functionality that is discarded in the iterated design.

The double meaning of the triangle with the exclamation mark has been solved by changing the fratricide icon (issue 16) and the symbol is no longer red in the iterated design (issue 24). During a future risk of fratricide and an acute risk of fratricide, the WDAs are still represented differently while in both cases they indicate a dangerous area (issue 19). However, this difference is used to quantify the urgency of this risk with yellow being a future risk and red being an acute risk. Therefore this difference is also kept in the iterated design.

The menu to access the alarm functionalities now opens up to the side, just like the rest of the menus in the interface (issue 25). Furthermore, an increase in size and opacity have been discarded as an indication of importance in the iterated design (issue 26). An increase of size is now just used in the salient alert and the change in opacity is discarded. Finally, when a section is selected it is no longer outlined in red (issue 27). As can be seen in image 6.9 the other sections get greyed out in the iterated design when the PC selects a specific section.

5. Provide feedback

The Raptor always responds to voice command in the iterated design (issue 28). When the PC gives the Raptor a command the Raptor will reply that he understood the command and tell the PC which action he is going to take (turning off the IED functionality for example). This is also more in line with D4SA principle 44 *"provide* *automation transparency"*.

7. Provide shortcuts

In the iterated design, when a visual alert is presented, the PC is always able to quickly return to the overview visualization by tapping somewhere on the map. This way the PC does not have to wait until the salient view disappears by itself (issue 29). The iterated design furthermore allows the PC to remove all the waypoints of a section at once, instead of having to remove them one by one (issue 30). It also becomes possible to hold an icon of a waypoint or tagged enemy and drag it around in order to move its location (issue 31).

8. Provide good error messages

The iterated design will let the PC know when it did not understand a voice command (issue 32).

9. Prevent errors

Issue 35 is discussed at the end of this section as it relates to a functionality that is discarded in the iterated design.

As can be seen in image 6.9, when a section gets selected in the iterated design the other sections will greyed out. This will make it more clear for the PC for which section he is placing waypoints (issue 33).

The iterated design will not show the PC how far a section can safely diverge from this waypoint (issue 34). This is done in order to prevent a clutter of the display.

Discarded functionalities

Two functionalities have been discarded in the iterated design. It concerns the functionality that presents a no-go area in order to prevent the risk of fratricide with another platoon, and the functionality that monitors whether a section ends up behind the enemy line. These functionalities were specifically related to claim CL002 and CL011, which therefore will be discarded from now on as well. Relatively a lot of issues related to these functionalities, which led to a re-evaluation of them. The reasons to discard these functionalities are discussed per function below, starting with the generation of the no-go zone.

During the specification of the system design, the functionality that generates a no-go zone in order to prevent friendly fire with another platoon surfaced as a result of a special use-case of the friendly fire predicting functionality. It was considered important information for the PC because it would help him to prevent a special kind of friendly fire without the need to contact the Squadron Commander (SQC). However, a few important factors about this no-go zone were missed, and these prevent a successful implementation of such an automatically generated no-go zone for the PC. First of all, the mission of the PC is part of a bigger mission. And, as became clear in the doctrine (chapter 4, section 6.2, source D), it is the responsibility of the SQC to prevent fratricide between platoons, and not the responsibility of the PC. The SQC determines the no-go zone on more information than is available for the PC. One example of this is that the SQC might be able to move the location of another platoon and open up other approach possibilities for the PC by doing so. With an automated no-go zone, these options are most likely not discussed between the PC and the SQC. More important is that this project did not include the involvement of the SQC. This means that his goals and information needs are not known. Inducing a change in the interaction and information flow between the PC and the SCQ could have unforeseen consequences. For that reason, the automatically generated no-go zone is discarded in the iterated design.

The functionality that determines the enemy frontline and checks whether sections end up behind this line surfaced from a special use-case of a section diverging from its route. From the interview with the former PC to evaluate the GDTA goal hierarchy (chapter 4) it became clear that a section ending up behind enemy lines is extra dangerous for this section. Therefore it was considered useful to be able to detect this specific kind of diverging from the route. However, this functionality requires an estimation of the enemy line. This information is not explicitly known by the Raptor and it is unclear for the designer how such a line could accurately be predicted. Leaving the estimation of the enemy front line to the judgment of the PC is therefore considered to be a better option. When the PC is responsible for this task, and he does not place waypoints behind his estimated enemy frontline, then he will still be warned when a section ends up behind this estimated line since the section will still move away from the waypoints in that case. The only difference is that the PC will not receive an alert that specifically indicates that a section is ending up behind the enemy line. Because of the difficulty of estimating this enemy frontline this function is also discarded.

6.4 Discussion

In this chapter, the system and visual design of the envisioned functionalities have been created, evaluated, and improved. Three methodologies have been combined together into a single process. This led to an effective and particularly quick development of a system that can be turned into a prototype. The different steps are first discussed separately, after which the process as a whole is discussed.

6.4.1 Initial design

During this first specification part of the process the focus was mainly on the system design. Putting the envisioned functionalities in context and thinking of all the different possible use-cases in which they could be used by the PC helped a lot in specifying how the functionalities should exactly work. Placing the system in all these different use-cases helped in making it more robust and prepared for unexpected scenarios. The downside of this focus on the system design is the fact that it focuses on textual descriptions so much. Appendix E shows the effect of this, which consists of many pages of use-cases, requirements and claims. This elaborated process can be a bit daunting at first.

Because of the large amount of work that went into specifying the system design, less time was spent on the visual design. The D4SA principles did provide helpful guidelines in the creation of the visual design and even contained suggestions on how to best visualize things like uncertainty or complexity. At the same time the D4SA guidelines were also useful for the system design since a part of the guidelines also related to interactions with the technology like keeping the operator in control.

6.4.2 Heuristic Evaluation

The SCE method promotes quickly testing concepts and hypotheses. For this purpose the HE showed to be very well suitable. The HE could be performed with four evaluators in a single day. This means that in eight hours 35 issues have been identified on which the next design efforts could be focussed. This is especially impressive since no target user is needed for this process. The heuristics were simple to understand for the evaluators and the power of the HE lies in the number of evaluators. As can be seen from the results, only 8 of the 35 usability issues were identified by multiple evaluators. Perhaps more issues would have been identified by multiple evaluators when more evaluators in total were used, which is also likely to increase the total amount of identified issues.

The difficulty of the HE is in assessing whether an evaluator indeed identified a usability issue or made a mistake. Interpreting the heuristics is of course a subjective measure, and in the paper in which the creators validated the method, there already were some arguments of subject wrongly identifying certain aspects of the interface as a usability issue [24]. At the same time it happened that an element of the interface was promoting a single heuristic, and in doing so violated another. An example of this is the structure of the menu to access the alarm functionalities. When this menu is fully opened it covers a significant portion of the display since it opens up to the side, what was identified as a violation of the simple and natural dialogue heuristic. However, this was done to stay consistent with the other menu structures of the interface, which were also opening up to the side, in order to promote the consistency heuristic. These contradictory heuristics limit the extent of the design improvement.

6.4.3 Iterated design

In the iterated design the focus was much more on improving the visual design than on the system design. Having the extensive list of identified issues helped a lot in focusing design efforts and quickly improving the design. The D4SA guidelines also helped in this process a lot to check some of the issues that were identified.

In some cases the D4SA principles were in line with the identified issue, like the issue that the Raptor did not communicate which action he was going to take after the PC gave a voice command. D4SA principle 44 stimulates automation transparency which means that the PC should be able to understand what happened after his command.

In other cases the D4SA principles were contradicting an identified issue. In these cases the issue was often neglected in order to benefit the SA supporting functionality of the related interface element. An example is the contradiction between issue 6, which was interpreted as the auditory alerts being lengthy, and the D4SA principle 27, which states that alerts should be unambiguous.

6.4.4 Entire process

The combination of the three methods seems to have had the desired effect of obtaining a usable interface that is aimed at increasing the SA of the PC. When the entire process is considered, the focus on the system design in the initial design is actually quite beneficial. Since the HE is focused on evaluating the usability, most of its identified issues will relate to the visual design, which means that a larger part of the focus goes to this visual design in the iteration phase.

The HE provided a useful role in the design process by defining specific elements to focus the next design efforts on. It is therefore also no issue if the initial design is made rather quickly. I would argue that it is easier for a designer to improve the design with the identified issues from a HE, than to improve it without these focus elements.

In the end, the combination of the SCE and D4SA methodologies is considered a design process that is very well suitable for a design and development process that is comparable to what is happening in this project. It creates a process that is both user- and technology-centered with a focus on improving SA. The functionalities complemented each other and by using the HE methodology a quick iteration on the design is possible. The end result is not only a visual and system design, but also a list of requirements that specify the required abilities that should be included in the prototype, as well a list of use-cases that provide the programmer with a lot of examples to showcase the reason of these requirements in more detail.

6.5 Conlusion

In this chapter the SCE, D4SA and HE methods came together to form a design process that exactly fits the needs of this project. It resulted in a process that is both user- and technology centered and is focused on improving the SA of the user. The SCE methodology was used to extract requirements and testable claims for the system design by creating multiple use-cases. The information that was collected in chapter 4 and 5 helped the designer in creating these use-cases, as the goals, considerations and information needs of the PC were well understood in these chapters. The D4SA method was used to provide the designer with SA specific knowledge that is lacking in the SCE method and was especially useful for creating the visual design. Finally, the HE was used to evaluate the initial design and focus the design efforts of an iterated design on identified usability issues.

The end result is a visual and system design that is further explained and put into context in a list of use-cases, requirements and claims. Together these can be used by a developer to turn the design into a prototype that works just like it was envisioned. The realisation of this prototype is treated in the next chapter.

Chapter 7

Implementing a prototype

7.1 Introduction

The goal of this chapter is to turn the design that is created in chapter 6 into a testable prototype. This prototype will be tested in a mixed reality environment to explore the effects of the new functions on the situation awareness (SA) of a Platoon Commander (PC) and in order to explore the added value in a practical scenario. The chapter answers the question RQ1.4: *"How can the created design be realized into a testable prototype?"* By answering this question, this chapter describes the creation of the artefact that was used for the final evaluation in chapter 8 with an experienced PC.

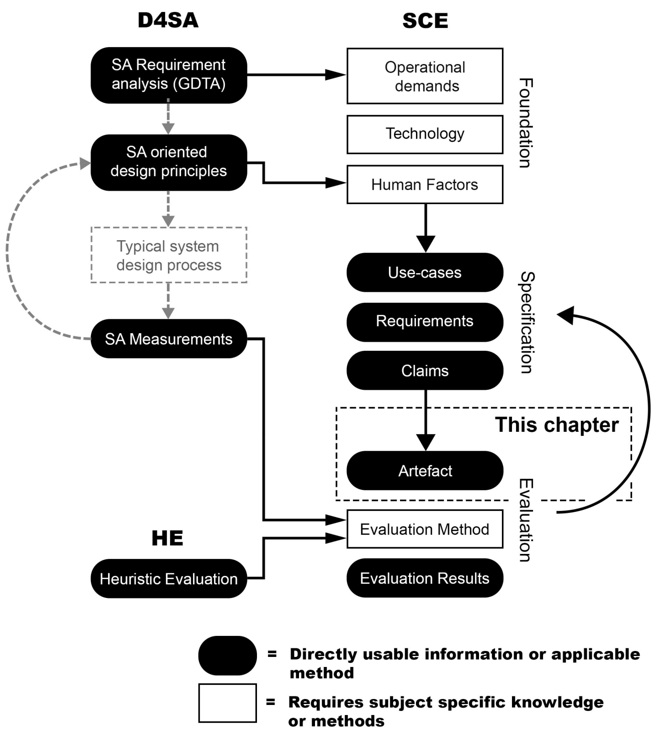


Figure 7.1: *Focus of this chapter with respect to the composite method*

To start this chapter, the method section is separated into (1) a section shows the environment in which the prototype will be used and (2) a section that defines the logic that needs to be programmed in the prototype for each functionality. The mixed reality and its simplified implementation of the Raptor are explained in more detail since it shapes the context in which the prototype will need to function. This simulation environment also includes the simplified version of the handheld that is shown in figure 7.5 and on which the design from chapter 6 is based. The logic that needs to be implemented is defined in order to better explain the problem at hand and what is needed to implement the functionality to the prototype.

The results section explains how the resulting prototype computed the logic for envisioned functionalities as well as some support functionalities following from the design in chapter 6.

In creating this prototype, a lot of functionalities were simulated instead of fully implemented. The design was not yet evaluated with the PC's, which means that it was unknown how the PC would evaluate the functionalities and their practical use. Therefore the prototype was created as simply and quickly as possible. The clearest example of this was the conversational agent that was needed for the voice commands. This could easily be implemented by using a human operator that behaves like the agent. The experience for the PC was not be changed by this: the handheld listened to the PC's voice commands. At the same time, the development time had seriously decreased.

To better illustrate the result, QR codes are used that link to videos that show how the different functionalities respond. It is possible to understand the entire chapter without accessing the QR codes, but the videos might make it easier to understand the results.

The prototype that was created in this chapter was used to perform a user-test with a former PC. This is presented in the next chapter.

7.2 Method

In this section, two major topics are discussed. First, the mixed reality simulation environment is treated in order to explain the context of the development process. After this, the logic problem for each functionality is defined.

7.2.1 Mixed reality test environment

The mixed reality environment for ground forces is developed by MultiSim [36], a company with which TNO works together a lot for executing experiments in simulated environments. MultiSim offers the ability to perform collaborative training of crews in joint simulation in a digital world that one can act in. Their software consists of a distributed database to which a wide variety of components can be connected. This ranges from the Unity rendering engine to a joystick or treadmill to move the location of a certain entity. The handheld is also connected to this database, called D-SIM. The mixed reality will be discussed in three sections: front-end, back-end and the old version of the raptor.

Figure 7.2: *Mixed reality setup for ground missions from MultiSim*

D-Sim front-end

As can be seen in image 7.2 the PC stands on a treadmill and sees the virtual environment projected around him. He can move through the environment by walking and using a small joystick to turn. Behind the PC you see two operators that are controlling the sections of the PC. They have the ability to move an entire section or individual soldiers. They can control their movement, speed, posture, and make them fire in a certain direction with a certain fire rate. The human operators also decide which units will be eliminated in a firefight. They can *"kill"* both friendly and enemy soldiers. The operator controls the elements in the simulation by using the D-SIM interface which is shown in image 7.3 and is presented to the operator on the lower screen. The upper screen shows the PC a perspective from the 3D world as shown in figure 7.4. This perspective can be the same as the perspective that the PC sees, but most likely this will be either the perspective from a section leader or a perspective in which the operator can move freely in the 3D world.

When the operator interface is considered in more detail we see the most important elements in image 7.3. This image shows the overview panel in the right part of the screen. Here the operator sees all the sections and their location on the map. He can select a unit and give a command like *"move to here"* or *"fire in this direction"*.

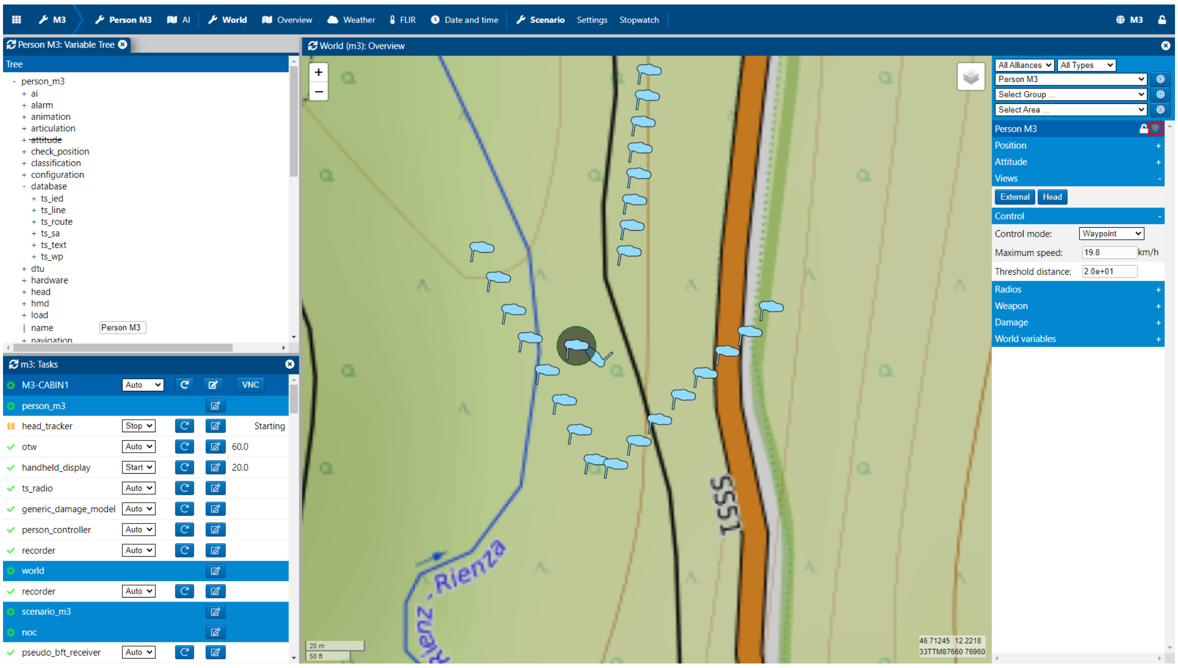


Figure 7.3: *D-SIM operator interface*

D-sim back-end

On the left side of the screen, we see two panels. The upper one is the variable tree. This panel lists the variables of an entity in the virtual world, M3 in this case (the entity of the PC) and their values. This panel is not supposed to be used by the operator during the execution of an experiment, but provides helpful information that can be used during debugging. D-SIM is a distributed database and using this panel the developer can access all the variables that are present in this database and check their status. Examples of this are the waypoints that are placed and the

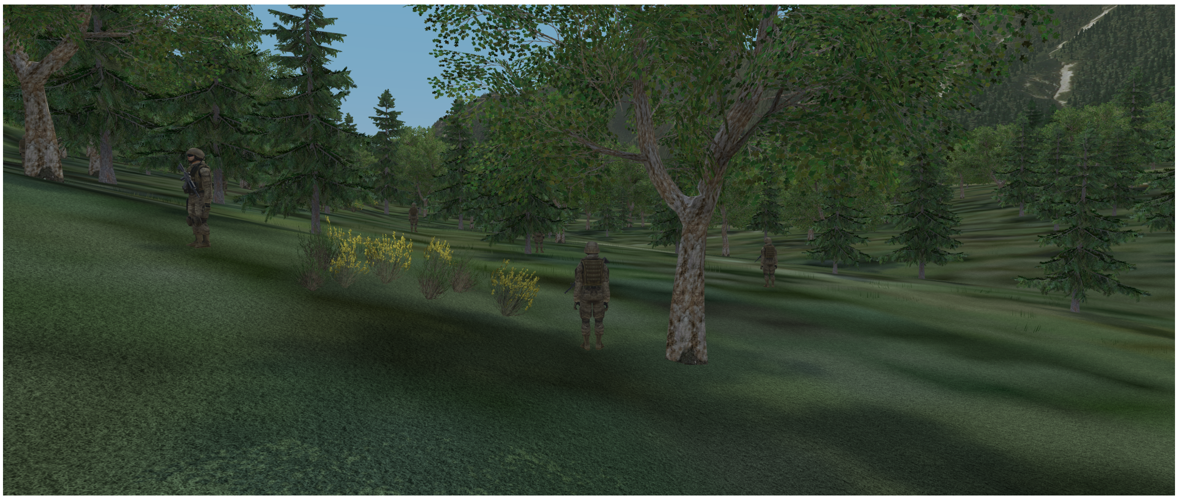


Figure 7.4: *3D environment that is projected around in the PC and that is displayed on the widescreen to the human operators.*

current state of the screen of the handheld. All information that should be available across different screens should be available in this database. This means that variables need to be added to this database when the new functionalities of the handheld use variables that will be synchronized between multiple handhelds.

The variable panel should not be used by the operator during an experiment because it lacks usability and it is very extensive: it contains a lot of information which is stored in a lot of different lists. If an operator needs to change these values manually he will be scrolling and searching for the correct information a lot. In order to solve this, it is possible to create a separate interface in which only specific variables are listed and can be edited. The interface will only present the information that is relevant for the PC, and handle the change in the D-SIM database when the operator changes a value. An example of this is the overview page. This panel is more user-friendly and only contains the variables that are needed to control entities in the world. Effectively, by using the overview interface, the operator is modifying variables in the database. These changes will then be visible in all applications that are connected to the database, as they pull this data for their execution. When a PC makes a section move to a certain location, the interface will slowly change the location variable of this section in the D-SIM database. The render engine and handheld application use this updated database again to create their visualizations.

The lower panel on the left side shows the task list. In this task list, different items can be connected to D-SIM. The handheld display is one of these tasks. Turning on this task will start the implementation of the handheld. Furthermore, the OTW task is visible, which starts the rendering of the 3D world. The *ts_radio* task updates the database between the different entities and tries to model the reality in doing so (latency etc. are incorporated here). It can be seen that this entity also contains a *head_tracker* task. This task is connected to an HTC Vive tracker that is mounted on the helmet of the PC as can be seen in image 7.2. This tracker is used to monitor the orientation of the head of the PC in three degrees of freedom (yaw, pitch, and roll). If another piece of hardware needs to be connected this can be done by creating a task for this or incorporating the hardware in another task. The last task that is visible in the task list is the pseudo blue force tracker. This tracker communicates the location of each section. The tracker updates every few seconds in order to model a realistic implementation.

The old version of the handheld

The final part of this mixed reality environment is the current implementation of the handheld. A video of the interactions and abilities of the current Raptor can be accessed using the QR in image 7.5. This version contains the basic interactions of zooming, swiping, and rotating, which are applied to the map.

Figure 7.5: *QR link to a demonstration video of the existing Raptor*

The PC has the ability to change the orientation of the map by clicking on the compass needle, and can pick from the options: north is up, the direction that the PC is heading is up, and free rotation. Furthermore the PC can center the map on their location by closing the lock icon, or allow themselves to move the map around by opening the lock icon. Finally he can also click a section and see its full name and any messages that it sent.

On the left the PC has two functionalities that he can access. The first one is the draw functionality, which lets him draw and remove lines from the map. By clicking on the filled square he can change the color of the line and by changing the pencil icon to the eraser icon he can either draw new lines or remove existing lines. The circle arrow allows the PC to remove his last line. The other functionality on the left of the display allows the PC to place waypoints in the map. The waypoints are numbered in the order in which they are placed. The PC can remove the waypoints only one by one by clicking them and pressing on the trashcan icon.

On the right the PC has two buttons that can be pressed. The chat icon allows the PC to open the messages tab. Here the PC can read all the sent messages and write a message him-/herself. By tapping on the map icon the handheld returns to the map again. The top button allows the PC to tap a location on the map and either mark it as danger in general or as an enemy location in particular.

7.2.2 Function definitions and required data and logic

This section will explain how all the functionalities are defined and which data en logic needs to be added to the prototype to make the prototype function as designed in chapter 6. This is done by first explaining how the functions are defined (when is a section diverting) after which the current available data is discussed. Finally the data and logic elements that are needed are presented.

First the functionalities that add the higher-order SA information to the handheld will be discussed. These functionalities originate from chapter 5 and their value for the PC can be read in section 5.3. After this, the functionalities that will allow for the interaction as designed and specified in chapter 6 are discussed.

Monitoring the movement of each section

A section was classified as diverting from its route, when the distance between its location and its route exceeds a predefined threshold. The PC uses waypoints to command movements to sections. By setting out consecutive waypoints for each individual section, a PC can set out a route, because the section will move from waypoint to waypoint. The information that the PC needs when a section diverts a moment is the extra time that it takes the section to reach the next waypoint.
In the current handheld the PC can only place global waypoints. These are waypoints for the entire platoon and are treated as individual waypoints. In image 7.6 for example, it is unclear if waypoint 2 and three form a route for section A or section B. This way the distance between a section and its route can never be determined.

The information that was available for these waypoints is:

- ID
- Location

Therefore, the following data elements were added to the D-SIM database:

- Waypoints should be assigned to an individual section
- Consecutive waypoints need to be combined into a route
- The predefined threshold.

And the following logic elements needed to be added to the handheld:

- Lines should be defined between consecutive waypoints that represents the route of a section
- The shortest distance between a section and this line needs to be calculated and compared with a threshold variable
- When the threshold is exceeded, the next waypoint on the route of the sections should be selected and the extra time to reach this waypoint should be calculated.

Figure 7.6: *Indication of the current situation with respect to the waypoints.*

Detecting and predicting acute fratricide

A risk of friendly fire occurs when a section is positioned in the weapon danger area (WDA) of another section and when no building or elevated terrain is placed between the two sections. A section is believed to have a risk of hitting any other section that is located within its effective weapon range, within its dispersion zone, and visible (so not obstructed by a building or elevated terrain). Being within the dispersion zone means that a section is located within the area that is defined by the safety angles left and right of where a section is aiming at. A future risk of friendly fire is predicted when a section is moving to a location where it will be at risk of fratricide. This happens when its current direction is moving the section to such a location.

In the existing version of the handheld the PC was able to see the location of each section and the movement of a section can be determined by looking at the change of locations. On the map used in the existing Raptor buildings were not visible, but a different map was available on which these building were visible. Elevation was not visible on the map. The speed of a section was not visible for the PC, but was available in the D-SIM database.

Therefore, the following data elements were added to the D-SIM database:

- The direction of a section
- The effective weapon range of a section
- The safety angles of a section

And the following logic elements needed to be added to the handheld:

- It should be computed whether a building or elevated terrain is located between two sections.
- It should be computed if a section is within the effective weapon range of one of the other section.
- It should be computed if a section is located within the dispersion zone of another section.
- When a section is not at risk of friendly fire it should be computed if it is moving to a location where it will be at risk of friendly fire
- When a section is moving in a direction where it will be at risk of friendly fire, the estimated time before the section arrives at this location needs to be computed.

Predicting IED locations

IED predictions were assumed to be made by a piece of software that was created by colleagues from TNO. A section will be classified as too close to an IED when it's distance to a predicted IED is lower than a predefined threshold.

In the existing Raptor IEDs were not present at all and could not even be added manually. This feature was therefore completely new.

Therefore, the following data elements were be added to the D-SIM database:

- Location of IED prediction
- Certainty of IED prediction

- The predefined threshold

And the following logic elements were added to the handheld:

- The distance between a section and an IED should be computed and compared with the threshold.
- When the threshold is exceeded the estimated time before the section arrives at the IED should be computed.

Tracking whether the PC looked at the handheld

In order to be able to follow the scheme in image 6.7 (chapter 6), the handheld needed to know when the PC was looking at the handheld. The PC was considered to be looking at the handheld when the PC was looking down.

The existing handheld did not take the gaze of the PC into account. However, the HTC Vive tracker that had been implemented in D-SIM before was able to monitor the orientation of the head of the PC.

Therefore no extra data elements needed to be added to the D-SIM database, but the following logic elements were be added to the handheld:

- The orientation of the HTC Vive tracker needs to be compared with a threshold to determine if the PC is looking down or not.

Understanding voice commands and creating audio notifications

This segment consists of two parts. The first part should be able to understand voice messages and turn off functionalities when the PC requests so. By understanding voice commands it is meant that the handheld responds correctly when the PC gives the command to turn a functionality on or off. Furthermore the handheld should be able to generate audio notification. By generating audio messages is meant that the handheld should be able to turn text into speech and that the audio notifications should contain the relevant information for that alert, like the section involved.

The existing handheld did not contain logic like this.

Therefore, the following data elements were be added to the D-SIM database:

- The status of a functionality (on/off)

And the following logic elements needed to be added to the handheld:

- Voice commands from the PC should be handled correctly
- The text for the auditory notification should be created and turned into audio

7.3 Results

The resulting functionalities are presented on a very global level in a set-by-step process. These steps could be implemented in other projects and in other languages as well while at the same time little coding knowledge is needed to understand them. The actual solutions that are visible in the videos to which the QR codes link are coded in C++ and OpenGL.

For each functionalities three things are listed: (1) used data from the D-SIM database, (2) data added to the D-SIM database, and (3) a step-by-step process of how this data is used and the logic that is added to the prototype. An image and a video also presented to show how the functionality responds in the prototype.

7.3.1 Monitoring the movement of each section

This part of the prototype is fully functioning. In order to achieve this, the waypoint placing interaction has been altered. The PC has to select a section before it is able to place a waypoint. This way section specific waypoints are placed that can be used to calculate the distance between a section and his route. How this function reacts to a section diverting from its route can be seen in figure 7.8 and via the QR code in image 7.7.

Figure 7.7: *QR link to a demonstration video of the existing Raptor*

The following data from the D-SIM database is used:

- Location of waypoints
- Current speed of a section

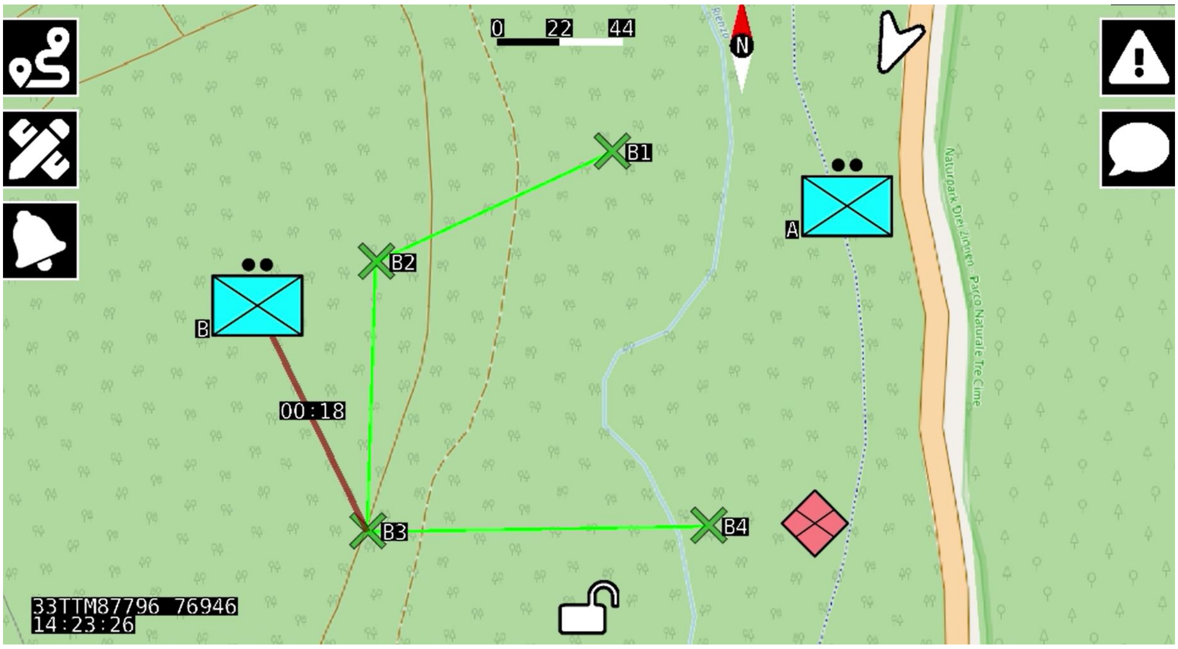


Figure 7.8: *Example of visual alert when a section is diverging from its planned route*

The following data has been added to the D-SIM database

- The section to which this waypoint is related.
- A route object that contains all the waypoints of a single section.

The following steps are taken to check if a section is diverting from its route:

1. Check for each section whether it has a route
2. When the route exists, the waypoints are treated as the points that make up a route. Two consecutive waypoints make up a segment of the route and a line is constructed between them. For each segment of the route a line is computed using the locations of both waypoints. The distance between the section and each segment is then calculated. For section S this means that the distance is calculated between S and the segment between waypoint 1 and 2, the segment between waypoint 2 and 3 and the segment between waypoint 3 and 4.
3. The shortest distance to the route is stored, together with the waypoint that is the endpoint of this segment. In the example the distance to the segment between waypoint 1 and 2 is the shortest, which will mean that distance d1,2 is stored, together with waypoint 2.
4. Once all the calculations on all the segments are finished, the shortest distance to the route is compared with the threshold. When this threshold is exceeded,

a line is drawn to the waypoint that is the endpoint of the related segment. In the example this means that the distance d1,2 is compared with the threshold, and a line is drawn to waypoint 2 when this threshold is exceeded.

5. To calculate how long it will take the section to arrive there the distance to this waypoint (waypoint two in the example) is divided by the current speed of the section.

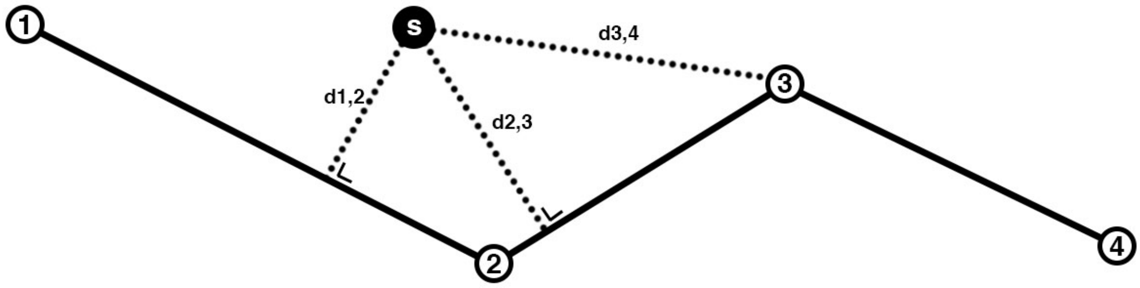


Figure 7.9: *Visualisation of how the shortest distance to the route is computed.*

7.3.2 Determining an acute risk of fratricide

This functionality is fully working in the simulated environment and can therefore be used in all missions that will be performed in this environment. The Unity render engine of the mixed reality environment is used to check if a building or elevated terrain lies in between two sections.

The handheld gives the location of the sections to Unity, which casts a ray between these two locations and returns a boolean that shows whether something is obstructing the path between the two sections (e.g. a building or elevated terrain). How this function reacts to a section being at risk of receiving friendly fire can be seen in figure 7.11 and via the QR code in image 7.10.

TNO

Figure 7.10: *QR link to a demonstration video of an acutre fratricide alert*

The following data from the D-SIM database is used:

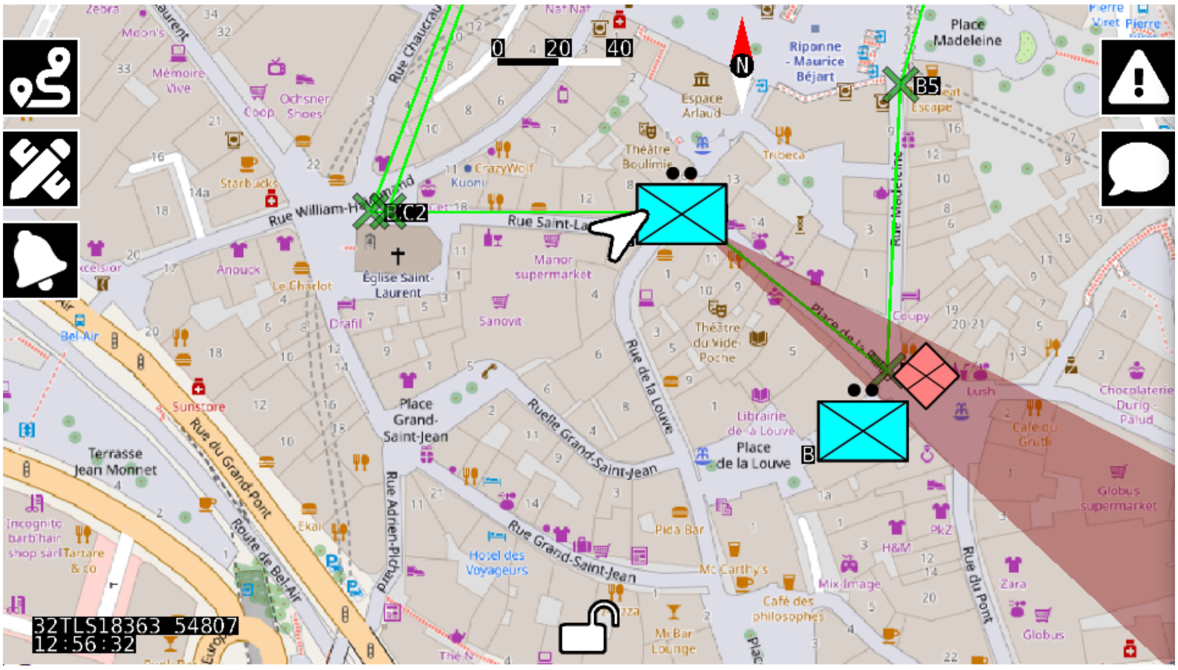


Figure 7.11: Example of visual alert when there exists an acute risk of friendly fire

- Location of the sections

The following data has been added to the D-SIM database:

- The obstruction boolean that is created by the performing the raytrace in Unity
- Effective weapon range of a section
- Safety angles of a section
- The orientation of a section

The following steps are taken to check if a section is diverting from its route:

1. Perform the ray tracing based on the location of all sections to check if a building or elevated terrain lies in between two sections. In the example image this would happen when a building or elevated terrain intersects line d1.
2. When the ray (d1 in the example) is not intersected, the distance between the two sections is calculated.
3. When this distance is less than the max effective range of the section's weapon this means that it could theoretically hit the other section. At the same time it is also checked if the distance between the sections exceeds a certain minimal threshold. This is done to prevent the fratricide alarm from going off all the time when the sections move as a single platoon and all the sections move

close together and look in the same direction. In such situations a section is within fire distance and in the WDA of another section, but the situation is not providing an actual risk of fratricide in practice.

4. When the sections can still theoretically hit each other, the orientation of the sections is taken into account. The angles between the sections are compared with the sections orientation. When the difference between these angles (a1 and a2 for example) is smaller than the safety angle (theta), it means that the other section lies in the WDA (which is not the case in this example, as the difference between a1 and a2 is bigger than the safety angle).

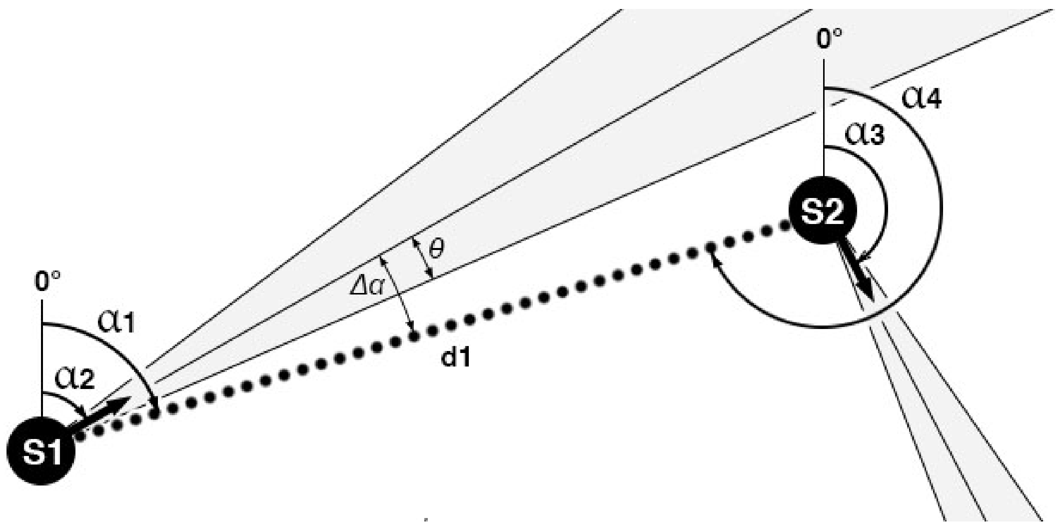


Figure 7.12: *Calculations for acute risk of fratricide*

The risk for future fratricide is not actually incorporated in the prototype. The limited available development time before the planned user-test caused this function to be neglected in the prototype. For the experiment, a static visualization of the alert has been made that fits in the scenario that is used in the experiment (which is presented in chapter 8) and can be seen in figure 7.13 to show the PC how the handheld would react.

If it would have been implemented it would have happened in a way that uses the calculation that is used to check for an acute risk of fratricide. For a future risk of fratricide, the location of all sections would be projected based on their current course and speed. For corresponding projections a fratricide calculation would be performed. Figure 7.14 illustrates how the portrayed situation would result in a future risk of fratricide in three timesteps ahead. Before that moment in time the buildings would block the risk of fratricide.

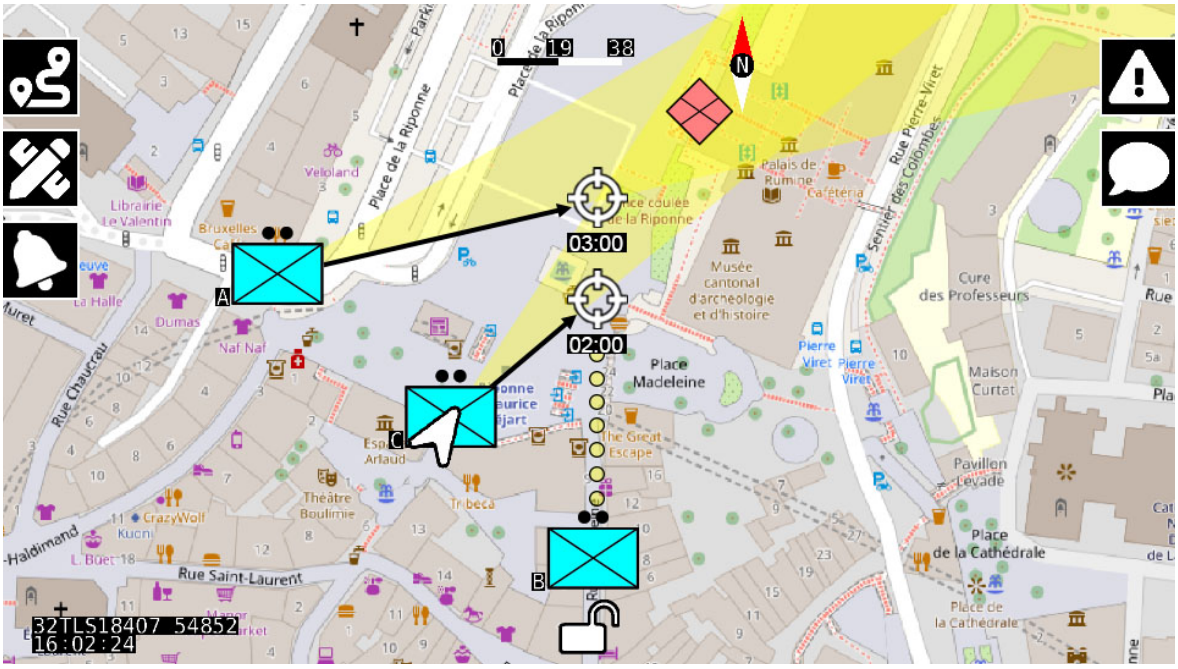


Figure 7.13: *Static visualization to show the visual alert for future friendly fire*

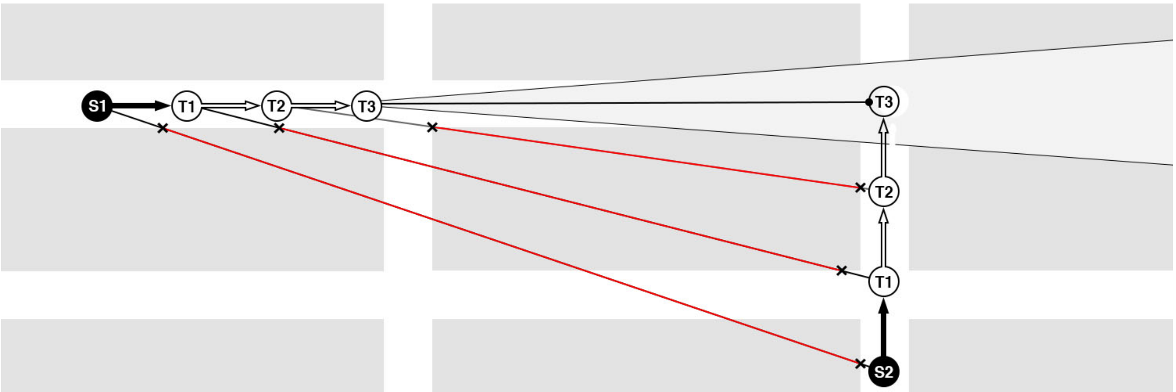


Figure 7.14: *visualization of how the future risk of fratricide would be calculated*

7.3.3 Prediction of IED locations

In the prototype the locations of the IEDs are not actually predicted. The human operator determines the location of IEDs based on the scenario that is executed. These locations are added as variables to the D-SIM database. The handheld then simply draws the IED icon on the corresponding location on the map. How this function displays the IEDs and reacts to a section getting close to a predicted IED can be seen via

Figure 7.15: *QR link to a demonstration video of an IED approaching alert*

the QR code in image 7.15 and in figure 7.16
.

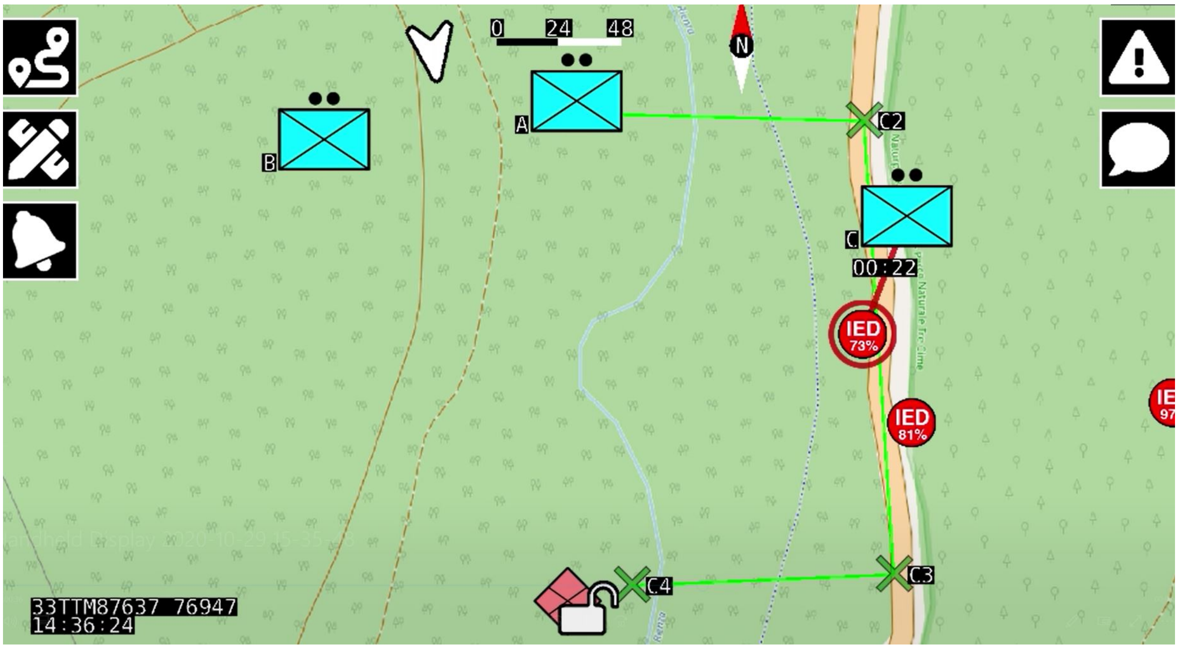


Figure 7.16: *Example of the visual alert that the prototype shows when a section approaches an IED*

The following data from the D-SIM database is used:

- The speed of a section.

The following data has been added to the D-SIM database:

- The predicted locations of the IEDS.
- Threshold value that represents a safe distance from the predicted IED location.

The following steps are taken to check if a section is diverting from its route:

1. Calculate the distance between an IED and a section. For example the distance between the IED and section S1 is d1.
2. Compare this distance with threshold t.
3. When the distance is smaller than the threshold, like d2 in the example, calculate the estimated time before the section reaches the predicted IED based on his current speed.

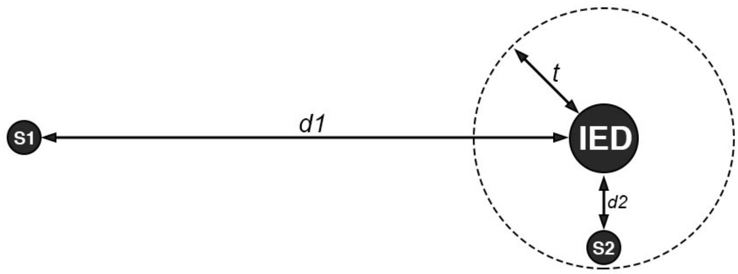


Figure 7.17: *visualization of how is determined whether a section gets too close to a predicted IED location*

7.3.4 Tracking whether the PC looked at the handheld

This support functionality is fully implemented and is used to determine whether a visual or auditory alert should be given. An HTC Vive control is mounted on the helmet of the PC. The handheld registers the PC as looking at the handheld when the angel of the Vive exceeds a threshold value alpha. These values are only used locally and are therefore not added to the D-SIM database.

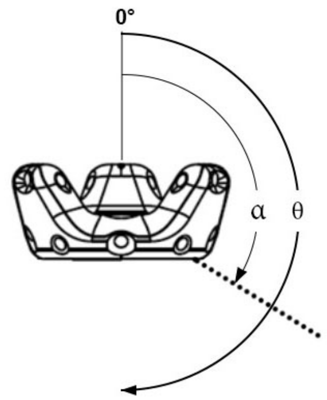


Figure 7.18: *Illustration of how the HTC Vive tracker is used.*

The following data from the D-SIM database is used:

- The orientation of the HTC Vive tracker (alpha)

The following steps are taken to check if a section is diverting from its route:

1. Compare the value of alpha with the threshold of theta.
2. When theta is exceeded, the time is stored.
3. Calculate the difference between the current time and the stored time
4. When this difference is bigger than 10 seconds the handheld will send auditory alerts. When the difference is smaller than 10 seconds, the handheld will send a visual alert.

7.3.5 Understanding voice commands and creating audio notifications

A simulated solution has been made to let the handheld understand the voice commands of the PC. In the operator interface a special panel has been made to enable the operator to handle the voice commands. He has tree checkboxes that show him which functionalities are currently turned on and off. When the PC gives the command to turn a function off, the operator simply taps this checkbox and the functionality will be turned off.

Figure 7.19: *QR link to a demonstration video of the use of voice commands*

In order to provide feedback to the PC about what the handheld is doing, and in order to give auditory commands, a TTS solution is used. In this case the simple windows TTS library for C++ is used to build a set of auditory commands that fit the scripted scenario. This solution could be extended in order to let it be responsive instead of simulated. In that case a template structure can be used for each alarm which the handheld can just fill in with the relevant info. For example: *"handheld to Romeo: Section [section] is moving away from its route in [direction] direction"*. The variables in the brackets can be replaced by the corresponding values.

Figure 7.20: *Interface by which the human operator can turn functionalities on and off*

7.3.6 Not implemented design elements

Due to time limits multiple aspects of the design from chapter 6 are not incorporated in the prototype. The focus during the realization of the prototype was on being able to receive feedback on the added value of the envisioned functionalities in a practical scenario. The design from chapter 6 also includes some modification on the current version of the handheld in order to improve the global SA of the PC. These modifications are not included and include the following elements: The new way of indicating the location of each section (including the movement vector), the historical locations of each section, and the difference in visual alerts between salient alerts and basic alerts. The latter was planned to be included in the prototype, but proved to be too time consuming after the basic logic for the three main functionalities and two support functionalities was completed.

At the same time, some parts of the design that more closely relate to the human factors and therefore the usability of the design are not implemented in the prototype. The following usability elements have not been implemented: The ability to select a single section and highlight the risks that are linked to this section, the ability to turn off a functionality only temporarily, and the ability to move elements like IEDs, tagged enemies and waypoints by holding their icons and dragging them. These elements can be added to the prototype once the functionalities proved to be of added value in the first place.

7.4 Discussion

The result of this chapter is a working prototype that functions as the artefact for the final evaluation of the composite method and can be used to perform the evaluation with two experienced PCs in chapter 8. Not all elements of the design that were presented in chapter 6 were included in the prototype. This is due to the limited amount of time that was reserved for the implementation of the prototype and is done to quickly iterate and validate the ideas of the current design. Since this design is very explorative, it is very likely that the functionalities will change substantially after the user-test, while a detailed implementation would take a lot of time. The design elements that are not incorporated into the prototype mainly consist of elements that improve global SA as presented in 6.3.1 and elements that relate to the usability of the design.

For this research project, the most important part is that the functionalities that include higher-order SA requirements to the handheld are included in the prototype in order to test the effects of these functionalities. A lot of these features are simulated to be able to test and validate them as quickly as possible. However, two important elements are not implemented, not even in a simulated way, in the prototype. The first is the difference between salient and global alerts. The second functionality is the prediction of a future risk of fratricide, which is also not implemented as the prototype can only detect an acute risk of fratricide. Especially the last one is important, as it is one of the core functionalities that adds higher-order SA requirements to the Raptor.

In view of the claims specified in chapter 6, it looks like almost all claims can be tested using the prototype with the functionalities that are implemented. The claim that relates to the detection of future risks of fratricide will be more difficult to measure (CL004). Chapter 8 will explain how this claim will still be tested. The lack of a difference between a salient and global visual alert is slightly affecting the claims that relates to the fact that the PC should not miss one of the mission critical cues (CL001, CL003, CL0011), but this can still be tested with the other alarm modalities that have been implemented. The prototype also meets the most functional requirements, with exception of the ability to detect a future risk of friendly fire (RQ005) and the ability for the PC to highlight all the alerts of a single section (RQ013).

A relevant question for this project is whether these not implemented functionalities can still be evaluated effectively. If it turns out that this is still easily evaluable, the question might be rephrased as to whether the result of this chapter indeed was the easiest way to be able to evaluate the design in a practical scenario. Perhaps there are possibilities to evaluate it with even less time needed for programming and by using more simulated elements or mock-ups. The next chapter discussed the user-test and will provide more insight into this question.

Finally, the role of artificial intelligence (AI) in this prototype is not directly visible. The goal of implementing AI to improve the SA of the PC lies in exploring ways of utilizing this technology instead of actually fully implementing and developing this technology. In different parts of the design the more machine-learning based AI is present and still none of these technologies are actually implemented in the prototype. The conversational agent and the IED prediction elements are the most clear examples. These elements form the technologically most complex part of the design. At the same time, these elements are the elements that are the most easy to simulate. In essence these technologies interpret something (be it a map of the city or a voice command) and based on this interpretation they perform and action. Simulating these technologies does not change anything about the information that the PC receives, but does reduce the time that is required to end up with something that can be tested single handedly. The combination of being technologically complex, but easy to simulate allows designers to experiment with them substantially. They offer great possibilities, while the costs (hours needed) to explore these possibilities are in practice much lower than one would expect initially. The experiment in chapter 8 will present the validity of this argument, but for now for now this approach, though arguable, is taken deliberately.

7.5 Conclusion

In this chapter the process in which the design of chapter 6 is turned into a prototype has been explained. The context in which this prototype has been realized has been introduced and within its boundaries a prototype serving as evaluation artefact has been specified that will be tested in chapter 8. The chapter shows a variety of fully implemented functionalities, simulated functionalities and not implemented functionalities. The approach of the realization is solely based on achieving a testable product as quickly as possible, from the idea that the design will be changed a lot after the evaluation after all.

The resulting prototype is fully able to monitor a section executing it's path. The prototype needs a human to predict the location of IEDs but is then able to inform the PC when sections move to close the assumed locations. The prototype is also able to detect an acute risk of fratricide, and takes buildings and elevation in account in doing so. The main functionality that is not included is the ability to predict a future risk of fratricide. Furthermore the conversational agent of the handheld is simulated. An operator can turn individual functionalities on and off when a PC commands so, and the handheld is able to generate spoken warnings when it detects that the PC is not looking at the handheld.

The implementation is running inside the mixed reality environment, which enables a PC to take place on the treadmill, lead a military mission, and use the envisioned functionalities in doing so. The handheld will respond to the unfolding scenario and notify the PC about mission critical cues. The PC is able to give the handheld voice commands to turn functionalities on or off and the handheld monitors when the PC has looked at the handheld in order to give auditory command at the right time. The next chapter will discuss the user-tests for which this prototype was developed in more detail.

Chapter 8

Testing the prototype

8.1 Introduction

This chapter describes the evaluation of the design with two former Platoon Commanders (PCs). In the composite method, this chapter describes the second evaluation and evaluation results of the design. The goal of this evaluation was to collect qualatative feedback from a subject matter expert on the design and implementation of the envisioned functionalities. Furthermore, the evaluation provided insights into how well the goal of this project (increasing the higher-order situation awareness (SA) of the PC) was achieved.

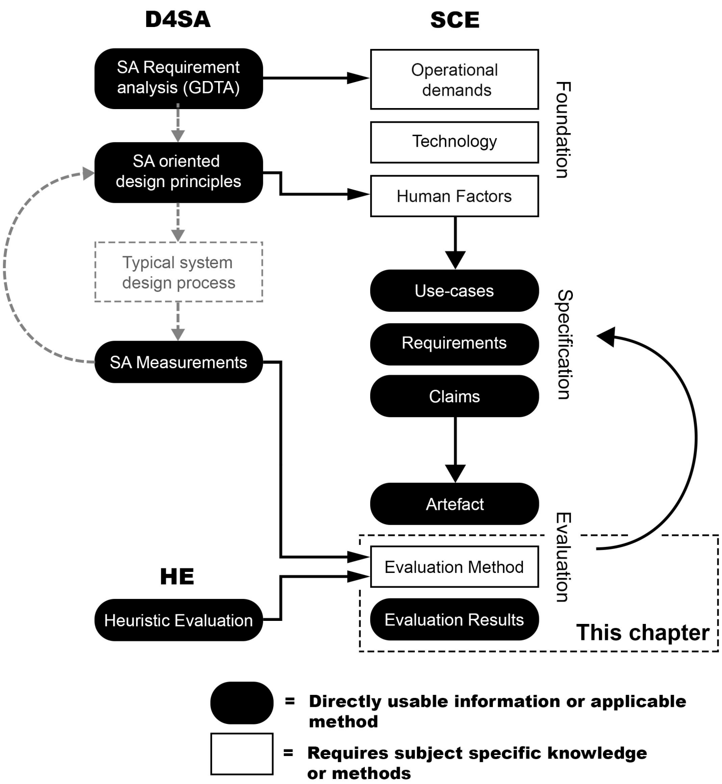


Figure 8.1: *Focus of this chapter with respect to the composite method*

To test the design, the former PCs observed the execution of a scripted and recorded hasty attack in an urban environment. The used scenario put the platoon in situations where the new functionalities of the handheld became apparent. This means that the sections approached improvised explosive devices (IEDs), diverted from their communicated route, and that acute- and future risk of fratricide occurred.

Due to tightened Covid-19 measures, the user-test had to be performed online. In this online evaluation, the PC saw the 3D world of the mixed reality environment on his own screen. The handheld was not constantly visible but was put on top of the view whenever it would be realistic for the PC in the scenario to take a look at the handheld (eg when he was standing still or when an alert was going off).

A soldier from the Dutch Marines that was working for TNO at the moment helped in the creation of a realistic scenario. The main functionalities of the design focus on three situations: sections diverting from their route, the occurrence of friendly fire, and the predicted locations of IEDs. The soldier indicated important features that can lead to a realistic occurrence of these risky situations. Using these features, a scenario was written and recorded in the mixed reality environment. How the user-test was performed and what the exact scenario looked like is discussed in the method section of this chapter. The remainder of this introduction is dedicated to introducing the elements that realistically cause one of the three mentioned situations to occur.

Diverting from route

There are three major reasons why sections divert from their route. The first one occurs in the case that the actual terrain looks different from what the PC believed it to look like when he set out the route. Examples of this can be destroyed buildings blocking roads, or a difference in terrain elevation. In these cases, the section is forced off to take an alternative route.

Furthermore, terrain that is difficult to navigate through is often selected for the approach of a flanking section. This provides them with cover from the enemy but at the same time makes it more difficult to monitor their location, which could also lead to a diversion from the communicated route. In this scenario, the section will be unaware of the fact that they are diverting from their route.

The last main reason for a section leaving its route is when this section encounters a stressful situation. An example can be an explosion nearby or when they take fire from enemy forces all of a sudden. In these cases, the soldiers try to take cover and might not only get disoriented in the process, but individual soldiers might even get cut off from the rest of the section. In this case, the section is most likely to realize only later on that they diverted from the route.

Strategic locations for IEDs

IEDs are generally placed on a point where they can have a big impact. On big roads or crossroads for example. Officially the locations of IEDs and minefields should be communicated to the enemy. However, especially when fighting militias, this will not be the case. IEDs can be used to either do damage or delay a military element, or they can be used to cover up a certain approach direction. Military personnel will look for digging activities to locate IEDs, and when on patrol they keep an eye out for seemingly randomly placed objects that function as a location indicator of the IED. When IEDs are placed on big crossroads, they are often placed just around a corner, in order to surprise the enemy even more.

Occurrence of risk of friendly fire

In general, there are two different kinds of friendly fire. The most obvious of the two is called direct friendly fire and occurs when a section is hit by the bullets of another section. This can happen when sections are not aware of each other's locations and get surprised by their allies. This can happen when a flanking section all of a sudden pops up in the region in which the fire section is providing fire, for example when a section unexpectedly moves around the corner of a building or moves up from a river bed. Situations like these should not occur, as sections should always communicate before they make such moves.

However, history shows that incidents like these do occur. Humans can make mistakes, and mistakes can be made more easily in the fog of war than during training scenarios. The adrenaline of soldiers that are ready to attack the enemy can sometimes lead to mistakes and elements like a smoke grenade can make it more difficult to distinguish friendly troops from enemy troops. The most well-known example of friendly fire in the recent history of the Dutch army happened when a Taliban section moved right in between two Dutch sections. When the sections shot at the enemy section, their bullets eventually hit friendly troops [37].

The second kind of friendly fire can occur when a call is made for air support. This can either be with mortars or a big 500-pound bomb that is dropped from an airplane. Indirect fire can occur when a friendly section is still in the area which is dangerous because of the direct impact of the bomb or because of the debris flying around as a result of the explosion. This happened to an American section in Afghanistan [38] for example.

8.2 Method

8.2.1 Participants

Two male subjects participated in this user-test. The goal of the experiment was to gather qualitative results about the created prototype. Both had been working as a PC in the past. One of two participants had more experience as an intelligence officer while the other had more years of experience as a PC in the Dutch Marines.

8.2.2 Materials

MultiSim's mixed reality environment for ground forces that was introduced in chapter 7 was used as a platform to test the SA supporting functionalities. In order to let the test subjects experience the handheld in the best possible way with limited human resources, an approach was chosen where the subject watched the execution of a scripted and recorded mission in the mixed reality environment. The handheld prototype that was created in chapter 7 was shown simultaneously to show the subject how the prototype responded to the unfolding scenario.

Figure 8.2: *QR link to a video that shows the stimuli during during the user-test*

The biggest urban environment that is present in the 3D world of D-SIM, modeled after Lausanne, was used to search for possibilities to create a realistic scenario. This was done by looking for elements in the terrain that could realistically lead to the mission critical cues (MMCs) as mentioned in the introduction. This scenario is visually represented in image 7.3, and the execution of this scenario by the platoon together with the response of the handheld can be seen in the video linked to by the QR code in image 8.2.

In the scenario, the PC was commanding a platoon that is setting out on a hasty attack. Their objective is to march to the green waypoint. Enemy presence is expected in the area, but their location is unknown. Whenever the platoon encounters an enemy unit, they are cleared to engage. The scenario is visually supported by image 7.3, which shows the different stages of the scenario. The panels are numbered and used in the textual description of this scenario to refer to the related panel in the following paragraph. A red circle surrounding a section means that it made contact. A red circle surrounding a section in combination with a red arrow means that this section is providing fire in the direction of the arrow. The full execution of the scenario, together with the response of the handheld, can be seen in a video that is linked to by the QR in image 8.2.

The scenario begins in (1) with a march. After a while, the platoon approaches a square. When the platoon reaches the edge of the square (2), the platoon makes contact with an enemy section. Section A takes cover and provides suppression fire. The PC makes a plan of attack and decides to let section B make a flanking movement.

During this flanking movement, section B makes contact with another enemy section (3). Section B takes cover and in doing so enters a different street, and moves in the direction of a suspected IED. The PC has to make a new plan and decide that section B can continue down this street as long as they approach the suspected IED location with care. The PC sends section C to the location where section B made contact with the enemy. Section C will provide suppression fire while section B is performing another flanking movement (4).

Section B runs over- and defeats the second enemy unit and continues its original flanking movement on the first enemy unit. The PC gives section C a new waypoint from where it can provide additional suppressive fire (5). Section B can surprise the enemy with their flanking movement, the enemy section is defeated (6) and all sections gather at section B.

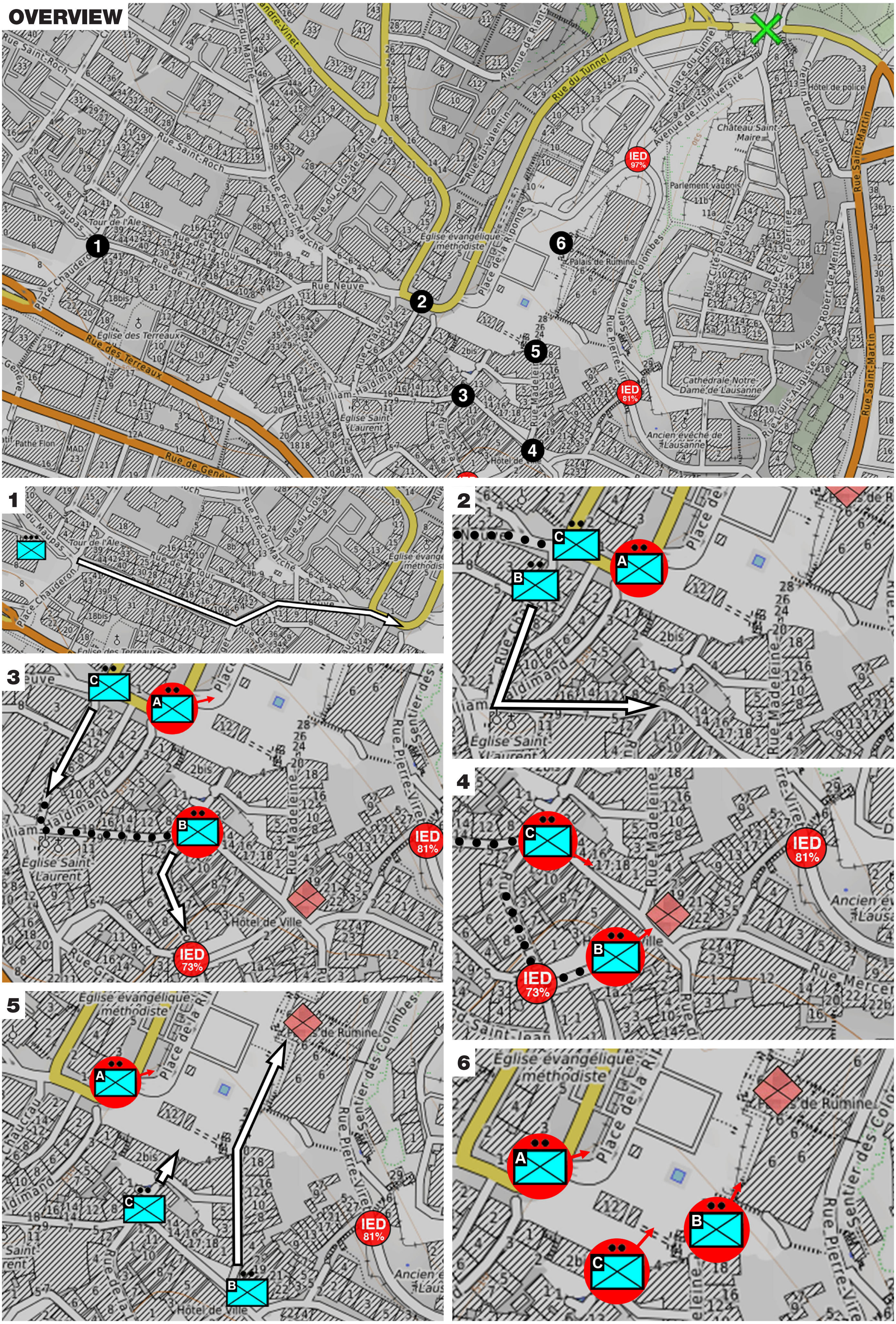


***Figure 7.3**: Visualization of the scenario that is used for the user test*

8.2.3 Measures

A semi-structured interview was conducted with the two subjects to collect feedback on the expected effect of the added functionalities on the SA of the PC, the added value of the functionalities in practice, and the usability of the current implementation. The questionnaire can be found in appendix G and consists of three parts related to the three focus areas of the user-test.

The first part of the questionnaire was used to access the expected effect on the SA of the PC. This part of the questionnaire is based on the MARS technique [26]. This is a verified, subjective measurement for the level of SA and the mental workload that was required for obtaining SA. Examples of questions from the MARS questionnaire are: *"Please rate how well (expected) the Raptor helped to identify mission critical cues in the mission."* and *"How much helped the Raptor in reducing the mental effort needed for identifying or detecting mission-critical cues in the mission?"*

Respectively the second and third parts of the questionnaire related to the usability of the current implementation and the added value in real-life scenarios. The questions for these last parts were self formulated and are used to collect feedback on the same topics in both sessions. Participants had to answer these question by stating how much they agreed with statements on a of a five point Likert scale. Examples of the statements are: *"The meaning of the alerts from the friendly fire predicting functionality was very clear to me"* and *"I believe that the function that predicts the risks of friendly fire in the near future will benefit a Platoon Commander on a real battlefield."*

8.2.4 Procedure

The user-tests with the two former PCs were performed in separate sessions in an online meeting. The evaluation started off with a small introduction of the test subject, the researcher, the decision support project, this specific research, and the procedure of the user-test. After the introduction, the researcher started sharing the screen of the mixed reality environment and the subject could see the 3D world. The handheld was not constantly visible but was put on top of the view whenever it would be realistic for the PC in the scenario to take a look at the handheld. These windows were not constantly overlapping, since the overlap would obstruct a lot of the view of the 3D environment. At the same time, this simulates the fact that a PC will also not constantly look at his Raptor in a real-life scenario.

The PC was asked to provide feedback directly instead of saving all his comments until after the scenario. The handheld is operated by the operator of the D-SIM interface. The scenario and the response of the handheld that the subjects saw can be seen by using the QR code in image 8.2. Every time an mission critical cue occurred, the scenario was paused and the researcher would give an explanation of the scenario and the reaction of the handheld to the subject. The questionnaire was then used to ask all the questions that relate to this mission critical cue and the answer would be written down. No video or audio was recorded and all comments from the subject were noted in a notebook.

Once the scenario was finished and the experimenter had asked all his questions, the experimenter would ask if the participants had any questions left, after which the subject was thanked for participating and the experiment was finished.

8.3 Results

The gathered feedback of both the participants is combined and grouped per topic. First, the feedback regarding the effect on SA and required mental effort is discussed. After this, the feedback on the current implementation of the three SA supporting functionalities is discussed, followed by feedback on the auditory messages and speech commands, feedback on the scenario, and feedback that does not belong in one of the previous subjects is discussed. Throughout this section, the claims made in chapter 6 are mentioned whenever the feedback relates to one of these claims. A summary of how the results relate to the claims is presented in section 8.3.8 and concludes the results.

In short, the test subjects were positive about the added functionalities and their implementations. They believe that they will increase the SA of the PC and make it easier for the PC to obtain this level of SA. The IED prediction functionality was criticized the most as it comes with the risk of creating a false sense of SA based on its predictions. Both participants also see added value in practice for the added functionalities.

8.3.1 Effect on SA and required mental effort

When asked to rate the effect of the new functionalities on SA and the mental effort that was required to form this SA, both participants responded positively. Both the participants think that the added functionalities help the PC in building SA. They both indicated that the functionalities help the PC the most in understanding and projecting the current situation (higher-order SA requirements). They even mentioned that the mental effort that is required to obtain these levels of SA is lowered by the added functionalities. One of the test subjects mentioned that the added functionalities provide the PC with very useful information about space and time variables. These variables immediately communicate the impact of a situation to the PC, which makes the information valuable. The PC on his turn needs to inform his superiors about the situation of his platoon. Directly understanding the impact on the mission allows the PC to easier communicate this to his superiors.

One test subject continued by adding that these functionalities are especially useful for the PC. An SC is able to rely more on his sight, hearing, and feeling where a PC is reliant on either communications or resources like this handheld device. The reason for this is that an SC is always located next to his section, while a PC is often located remotely from one or more sections. Therefore the same test subject mentioned that SA supporting functionalities definitely add new information to the handheld that will better allow the PC to understand the situation, by for example predicting future risks of friendly fire.

One of the test subjects illustrated the added value of the SA supporting functionalities by comparing them with road navigation. A driver can find the correct route by looking at the map himself and estimating his arrival time based on the remaining distance and his current speed. However, road navigation will make this a lot easier for you. At the same time, it will also continuously check whether you are still driving on the correct route and keep updating your expected time of arrival. When translated to the military domain, the subject mentioned that this increase in information density significantly reduces the *"sensor to shooter"* time, which can be explained as the time between receiving a signal and taking an action. This also explains why both participants mentioned that they expect that the mental effort required to obtain good SA was lowered by the functionalities that have been created in this project.

Finally, one of the test subjects explained that it can also happen that the PC's only connection with an SC during the execution of an attack consists of a few radio communications. He mentioned that all the added SA supporting functionalities, therefore, transform the PC from blind to partially sighted. At the same time, the same test subject sees good opportunities to incorporate future technologies with the added functionalities. Examples are an eye in the sky or eagle eye drones, which can quickly scan the area on contact and are able to detect dangerous elements like IEDs. These elements could be drawn on the map and compared with the location of the sections in the same way as is done for IEDs in the prototype.

8.3.2 Implementation of the functionality that monitors the movement of each section

Both test subjects indicated to recognize the added value for the PC in the functionality that checks if sections are diverting from their route. They stated that it enables the PC to understand the current situation better, even at a single glance. This assured that the PC will not miss sections diverting from their route and therefore confirms CL001. The subjects considered this especially useful in a similar situation like the one that was used in the scenario, where the section diverted from their route because they made contact with the enemy. In a situation like this, the priority of the SC is to keep his section safe. This means that he won't have time to thoroughly communicate the situation to the PC. With this functionality, the PC is already informed about the change in movement from the SC and the impact on the mission. Even though he might not know the reason, he can already start to think about the consequences and communicate to the fire support section that they need to be able to keep providing suppression fire longer.

One of the test subjects mentioned that he liked the fact that the functionalities can be turned off individually. This confirms CL009 that which must assure that the PC stays in control of which information he receives. The subject even mentioned that he would like to be able to turn them off for individual sections in order to have more control over the information that he receives or wants to neglect. As an example, the subject added that it should not be forgotten that an SC can always divert from his commanded route when he believes that this is needed. The SC can judge the environment with his own eyes, and it might look different from what the PC expected. The ability to turn off the alerts for this functionality allows the PC to prevent the alerts from continuously going off in such a case. The same subject mentioned that it still remains useful in this case for the PC to know where the section is located when the SC changes the route and what the change in time of arrival at the enemy's position will be and that the functionality can help with that.

8.3.3 Implementation of the functionality that monitors the risk of acute and future fratricide

Both test subjects indicated to believe that this functionality adds value to the PC. They both argue that the predictive feature makes it a lot easier for the PC to prevent a risk of fratricide, especially when an attack is executed over longer distances.The fact that the PC can directly see the time before a risk of fratricide will occur plays an important role in this observation. This comment supports CL003 and CL004 that state that a PC should be assured to never miss a (future) risk of fratricide.

For the acute risk of fratricide, one of the participants commented that the timing of the alert is essential. When a section is experiencing a friendly fire, the SC will notice this because they notice the bullets flying by; i.e. when they notice, it is already too late. The test subject, therefore, suggests incorporating a buffer in the alarm: 10 seconds before this section enters a WDA, the alarm of an acute risk of friendly fire should go off. This way the section should immediately stop moving forward and the fire support section should also immediately stop shooting.

For the PC this information is very useful again, as he is not located at the location where the risk occurs. The SC is able to hear the suppression fire, or see the bullet impact. The PC again is reliant on second-hand information, which the handheld is now providing him. At the same time, the stakes for the SC are perhaps even higher than for the PC. When the SC misses this warning he can lose his own life and that of his entire section. The PC's own life will not be directly at stake during an occurrence of friendly fire, although of course, the PC will do everything in his power to not lose his men.

One test person mentioned a situation that should be included in the current implementation to limit false alarms even further. The subject was questioning whether the current implementation allows a section to provide cover fire over the approaching section. When elevation difference allows for this approach, like a ditch or a valley, the fire support section can provide suppression fire by firing right over the flanking section, without an alert for friendly fire being issued.

Finally, one test subject mentioned that it is important to make the information about the risk of friendly fire also available for other platoons and the PC's superiors. It can happen that the movement of one platoon can cause a risk of friendly fire with another. When this information is available for the other platoon, the callsign of the section providing the risk should also be available in order to contact this section as quickly as possible. In order to prevent friendly fire, this platoon could take cover and wait on communication with either the Squadron Commander or the PC of the other platoon.

8.3.4 Implementation of the functionality that predicts IED locations and monitors these locations

The current implementation of the IED predicting functionality is perceived as less effective. As expected, the subjects immediately questioned the technology behind the IED predictions. After the research that achieved good results in predicting these locations was mentioned, the opinion of both subjects was that this information (if indeed possible to obtain) will be useful for the PC, especially because the handheld is also monitoring whether sections are approaching the IEDs.

However, both test subjects remain skeptical about the use of predictions of IED locations. This also has a more psychological reason; when predictions are made of locations for IEDs, humans tend to automatically think of other locations as IED free. One of the former PCs had experience as an intelligence soldier and made the IED predictions for multiple platoons himself. This subject mentioned that he had to keep reminding the soldiers to which he provided the predictions to keep their eyes open, also on locations where he did not predict an IED. This subject said that providing the soldiers with predicted locations of IEDs can easily create a false sense of SA, for which everyone has to be very careful. These statements create an interesting discussion around claims CL005 and CL006 that state that the handheld will help the PC invalidating his own prediction of IEDs and create a plan that anticipates the presence of these IEDs.

The difficult thing with both claims is that the test subjects indicate that the IED predicting functionality will create a false sense of SA. This means that the IED functionalities can have the claimed effect on the PC: thinking that the plan avoids IEDs because it avoids location predicted by the Raptor and having the sense of validating the PC's own predictions. However, as mentioned by the test-subjects, these effects are likely to be based on a false sense of SA. Just like the IED functionality, the claims therefore also need to be changed to accommodate for the risk of this expected false sense of SA.

Both subjects mentioned that they would have liked to see the prediction of locations of IEDs be replaced with the identification of key areas. The subject says that both parties (friendly and enemy) are interested in the same areas when fighting. These areas provide good lines of sight (along which soldiers could also fire) and high areas or buildings. When a section is moving to an area that is not yet secured, it is very likely to make contact with enemy forces on areas like these. When these areas are known, and the lines of sight/fire are drawn on the map as well, then this could help the PC a lot in determining a suitable approach for the area as well. The false sense of SA might be less big of a deal here and these key areas seem fit to be interpreted by AI technology as well. It will always remain the responsibility of the PC to check the prediction, but the prediction could make it easier for the PC to quickly identify the areas in the first place.

Finally, one of the subjects mentioned that the MoD is focussing on fighting against near-peers (enemy of almost equal strength), which are less likely to use IEDs, which reduces the practical added value of this functionality. At the same time, near-peers are more difficult to fight, which makes it even more important to correctly understand the terrain. A Raptor that identifies key areas on a map might help the PC with this.

8.3.5 Using speech commands and auditory notifications

Both participants were positive about the possibility to receive auditory notifications about important risks. However, the implementation of the auditory messages received some criticism. Right now they are quite lengthy. In military communications, there are different ways of communication for different levels of risk that the platoon is currently experiencing. When these auditory notifications are given, the platoon will be at risk, and because of the urgency of the message the notifications need to be very brief. The PC will always have too much information available to continuously process. Therefore the auditory notification should be as short as possible, the so-called extra shortened communication protocol. Furthermore, auditory messages need to use unit-specific language. For example, the auditory messages in the prototype reply with wilco (short for will comply). One of the test subjects (a former member of the special forces) mentioned that the special forces never use wilco, but use roger instead.

8.3.6 Scenario

One topic that was not included in the questionnaire that was used to guide the interview, but was criticized in both user-test, was the scenario and execution of the scenario. First of all, the way the platoon is moving is unrealistic. In a realistic scenario, a platoon would never march in the middle of the street. They would march on both sides of the street with spacing in between individuals so group members are able to provide cover. At a large crossroads, the platoon would also never just move around the corner. The PC will always send some scouts forwards to check for danger after which the platoon follows.

Another comment that was made was that a platoon in an urban environment would never cover such large distances on a single mission. The urban environment provides a lot of possibilities for danger. This means that risks are high, movement is slow, and covered distances are small. A PC will let his sections enter buildings, especially when he knows that the enemy is likely to be present. One of the test subjects mentioned that a mission like this, in an urban environment, is most likely not even performed by a squadron. The chosen scenario is much more realistic in a rural environment.

Furthermore, a PC will never move on his own in a city. Even though this possibility was given for a mission in a rural environment, one of the participants mentioned that a PC will always stick with one of his sections in an urban environment. The reason for this is the involved risk, which is just much higher in an urban environment than in a rural environment.

8.3.7 Other feedback

One of the subjects mentioned that it would be useful to show to the PC how a section moved to its current location and in which direction it is currently moving. This helps the PC to predict the future state of the situation. Furthermore, the same test subject highlighted the importance of sharing information between different layers of command. The clearest example of this is that the information about the risks of friendly fire is equally important for other platoons as for the sections of a single platoon since friendly fire can also occur between different platoons or even squadrons.

The participants also provided some further insight into military action in an urban environment. When a PC is preparing a mission he will always study the following three elements: weather, terrain, and the enemy. Furthermore, not only the covered distances are different between a mission in a rural area and an urban area differ; the rules of thumb to engage an enemy differs as well. Where outnumbering an enemy by three to one will suffice for opening fire on an enemy in a rural area, this increases to one to eight, or even one to ten in an urban area. Finally, one participant mentioned the importance of the gut feeling of commanders during warfare. The subject indicated that it was not the superior intelligence that made the difference between losing or winning battles in for example the war in Afghanistan, but the *“7th sense”* of commanders. This *“7th sense”* is triggered by much smaller variables, like the position of the sun and the presence of children on the street: When the PC has to look into the sun during a fight, this is beneficial for the enemy, making it more likely for them to attack. When there were no children on the streets, their parents were probably aware of a planned attack and kept their children inside.

8.3.8 Overview with respect to the claims

Table 8.1 shows an overview of all the results with respect to the claims and takes a closer look at claims for which the results allow for some discussion. In an earlier iteration of the design, the parts of the design that related to CL002 and CL011 were discarded and therefore these claims will not be discussed here.

Claim	*Description*	*status*	*Comments*
CL001	*The Raptor reassures that the PC will be aware of sections that diverge from their planned route*	*confirmed*	Notifications can be understood and the presentation conveys the impact of the MMC at a single glance.
CL003	*The Raptor reassures that the PC will be aware of occurrences of acute risks of friendly fire*	*confirmed*	Notifications can be understood and the presentation conveys the impact of the MMC at a single glance.
CL004	*The Raptor reassures that the PC will be aware of future risks of friendly fire*	*confirmed*	Notifications can be understood and the presentation conveys the impact of the MMC at a single glance.
CL005	*The Raptor reassures that te PC creates a plan that anticipates on the presence of IEDs*	*Needs to be modified*	Strictly takes, these claims are confimred, but the effect will occurs on a false sense of SA. The subjects expect that these functionalities the PC less attentive to these IEDs. To be of pratical use the claims needs to accommodate for the risk of this expected false sense of SA.
CL006	*The Raptor reasures that the PC validates self made predictions of IED locations*	*Needs to be modified*	
CL007	*The Raptor reassures that the PC does not receive redundant notifications*	*not tested*	User-test was not suitable to test this but the subjects did mention that the amount of alerts in the shown example was not redudant.
CL008	*The Raptor reassures that the PC stays in control of which information he receives*	*confirmed*	One subject mentioned that even more control is prefered and would like to be able to control every functionality for each section specific
CL009	*The Raptor reassures that the PC will always able to interact with the Raptor.*	*confirmed*	Auditory alerts need to be communicated following the extra shortened communication protocol
CL010	*The Raptor reassures that the PC will not miss a section approaching a predicted IED located.*	*confirmed*	Notifications can be understood and the presentation conveys the impact of the MMC at a single glance.

Table 8.1: *Overview of how the results relate to the claims*

8.4 Discussion

The results of the user-tests show that the designed functionalities are received mainly positively. Both test subjects indicate to believe the functionalities will add value to the PC in a practical scenario, that they will increase the PC's SA and make it easier to do so. In general, this means that it is advised to continue the development of these functionalities for inclusion in the Raptor. The results of the user-test already show a lot of ways to further improve the implementation of the functionalities.

The results show that the functionality that monitors the movement of the sections was very well received. The only suggestion that was made was to further improve this functionality was to enable the PC to turn off this functionality for a specific section. It might be useful to consider this in the next iteration, although it might also require a more complex interface structure to control each functionality for each single section. The challenge is to keep it clear for the PC which functionality is turned on for which section in that case.

The functionality that predicts the future risk of fratricide between sections was also very well received. A suggested addition to this functionality is to include other platoons and even squadrons in the calculation as well. This will require further analysis of how this information should be presented at these levels and what information they exactly need. It is likely to differ from the information that the PC needs, as other levels have other goals and decisions to make. The functionality that monitors for an acute risk of fratricide is also very useful but the timing of the alert should be altered. By giving an alarm ten seconds before a section moves into the weapon danger area (WDA) of another section, actual friendly fire can still be prevented. When the alert is only given when friendly fire is already occurring, the notification is already too late.

The most criticism was given on the IED predicting functionality. This functionality was still considered to be useful, but the downside of the marked locations of expected IEDs is that it can create a false sense of SA. At the same time, the relevance of IEDs is decreasing as the MoD is focusing on fighting near-peers that are not likely to use IEDs. This functionality might add more value to the PC when it is used differently. The functionality could be used to predict so-called key areas, that consist of locations with good lines of sight/fire and often include elevated terrain and high buildings. It is clear that the IED predicting functionality needs to be reconsidered and if the use of it will be changed to the interpretation of key areas, this function has to be designed from the ground up.

The auditory notifications need to take the specific language used by different sections of the army into account. These differences need to be implemented correctly in order to create a satisfying interaction. At the same time, there are also big differences between the environments in which a mission is executed. These differences have implications for the added functionalities. It would be useful to keep their functioning flexible. Enabling a PC to change certain thresholds, for example, could enable the handheld to be useful in all scenarios as we have seen that distances differ per environment. A PC could change these settings during the preparation of the mission, but this prototype did not include ways for the PC to change these settings.

At the same time, the comments of the users should not be taken as a holy truth. The user-test was performed with only two former PCs, in an environment that was not really interactive after all. The result should be interpreted more globally. The main implication of the user-test is that it seems to be worth it to invest more time and resources in fully developing these functionalities, something that remained a question until the user-tests. The important information here is that both users were positive about the added functionalities to the handheld. These results should definitely not be interpreted as that the implementation of each function is perfect in areas where no negative feedback was given on the implementation of certain functionalities. More elaborated and quantitative testing is required to find out in which way the actual design and implementation of the functionalities can be improved.

At the same time, the user-tests also provide interesting insights into the testing procedure of these kinds of designs of new functionalities. The best insight comes from the evaluation of the functionality that predicts future risk of fratricide. This functionality was not included in the functional prototype but was only presented to the test subjects as a still, at the corresponding time in the scenario. Nevertheless, the test subjects were able to give feedback about this functionality equally as well as on the other functionalities. This observation is not confirmed with the test subjects but is based on the experience from the researcher during these evaluations. During the semi-structured interviews, there was no difference between discussing the future fratricide predicting functionality and the implemented functionalities. The true power of the functional prototype was not used because of the last-minute need to conduct the experiment online. The fact that a not fully interactive scenario could also be used to perform the user-test and that this already yielded a lot of useful feedback shows that less time could have been spent on the creation of the evaluation artifact, as it is more interactive than was necessary.

The easiest way to evaluate the design would have been to evaluate it in the same way as the future risk of fratricide has been evaluated right now. The scenario will still be recorded in the mixed reality environment, just like the response of the handheld. The design can then be applied to the occurring situation in a series of stills, or their visual appearance can be programmed and triggered by a switch, leading to a non-responsive prototype. This would save the coding of the logic behind the prototype, which is by far the most time-consuming part. The power of the functional prototype is its ability to respond to multiple situations. In this stage of development, this level of flexibility was not necessary to collect relevant feedback on the design. However, in later stages, this might be necessary, and at that point, a functional prototype will be very useful.

8.5 Conclusion

The conclusion of the user-test is that the reactions from the PCs are positive enough to continue the development of the added functionalities and eventually include them in the Raptor. At the same time, the functionality that predicts the IEDs should be reconsidered. It's (future) relevance should be re-examined and the risk of providing a false sense of SA should be taken away or at least limited. At the same time, it is recommended to look at another use of the functionality, namely to identify so-called key-areas.

The final implementation should pay close attention to the details that play an important role in the final implementation of the functionalities. These details entail both the relevant army unit that is using the Raptor as well as the environment in which the Raptor is used, whether that is an urban, rural, or another environment.

The user-test shows signs that the added functionalities indeed increase the SA of the PC and also make it easier for the PC to obtain this SA. In more detail, both participants indicate that added functionalities add the most value for comprehension and projection of the current situation, which are the higher-order SA levels and was exactly the goal of this project.

The final conclusion of the user-test is that the design could have been tested in a more easy way. Where it was thought that an operational prototype was needed to collect good feedback, this proved to be not even necessary. The functionality to predict the future risk of fratricide could also be evaluated. Evaluating possible new functionalities, in the same way, will save even more time in the development process.

Chapter 9

Discussion, conclusion and future work

9.1 Introduction

This chapter reflects and concludes on the design and development process that was constructed for this project as presented in chapter 3. The results of the user-tests, presented in chapter 8, are used to do so. Section 9.2.1 starts off the discussion section by evaluating the composited method and discusses concepts that relate the results of the user-study to the areas of artificial intelligence (AI) and decision making. The conclusion of this research project is presented in section 9.3 and concludes the research questions and the answers that have been formulated in the previous chapters. Finally, section 9.4 discusses possible ways to continue with the design and implementation of the functionalities that were created in this project.

9.2 Discussion

9.2.1 The used methodology

The composite method presented in chapter 3 was used to answer the main research question *RQ1: "How can the current version of the Raptor be enhanced with higher-order situation awareness components to increase the situation awareness of the Platoon Commander during a hasty attack?"*. This subsection first briefly recaps this composite method and continues to the lessons learned from the process. Lastly, some remarks and made on the used method.

Brief recap

This research project's goal was to specifically add higher-level situation awareness (SA) components to the Raptor. Previous research does not provide a methodology that is aimed at designing and developing these components in particular. At the same time, the hypothesis of the project was that higher-level SA requirements could very well be generated by AI technologies. The strengths of AI technology are classification and prediction, which are in line with the kind of information that is represented by the higher levels of SA: comprehension and projection.

In order to still be able to add these higher-level SA requirements in particular, this research project designed, used, and described a methodology that consists of a combination of existing methods: Socio-Cogntive Engineering (SCE), Design for situation awareness (D4SA) and the Heuristic Evaluation (HE). The existing methodologies are combined to incorporate all the different aspects that are important in designing to support higher-level SA. The resulting composite methodology was already presented in chapter 3 and focuses on the following three concepts: SA, Human Factors, and AI.

The SCE method is used as the main structure for the composite method. The D4SA method provided both a method to better understand the needs and consideration of the user (used in chapter 4 and 5), substantial SA oriented design knowledge (used in chapter 6), and a method to measure the level of SA of a user (used in chapter 8). The HE provided a method that allowed a first evaluation of the initial design without the involvement of the target user (used in chapter 6). The combined methodologies provided excellent ways to understand the user, to put this knowledge in context, and to quickly test ideas and make iterations.

Lessons learned

The added value of a created composite method relates closely to the need for more research on level 3 SA that is mentioned by Wickens [39] and Endsley [40]. Being able to specifically create functionalities that will meet these higher-level SA requirements could benefit their researchability. The results from the user-tests at least show promising results for this composite method. Not only did the test subjects indicate to believe that the added functionalities increase the higher-order SA levels of the Platoon Commander (PC) in particular, but they also indicate that they help PCs to build their SA more easily since they made the current situation understandable at a glance (level 2 SA) and indicated the impact on the rest of the mission (level 3 SA). For this reason the created composite method might be useful for other studies and design and development projects that are particularly aimed at higher-order SA components. However, no hard conclusions can be drawn on this only involved two participants.

The user-test results show indications of how well the composite methodology helped in creating a good design and implementation. The best example of this is the fact that some parts of the design that was created in chapter 5, but, due to time limits, could not be realized in chapter 6, were mentioned as a suggestion by the test subjects during the user-test. The most clear example of this is the trail of previous locations of a section that indicated how it reached its current location and a movement vector that indicated in which direction this section is heading. The subject mentioned that these elements would make it more easy for him to predict the movement of these sections and these design elements were a direct result of the SA oriented design principles and design method and were never shown to the test subject. That they these elements were mentioned shows how well this composite method is able to create an effective and efficient solution and allows the designer to understand the needs of the user.

At the same time, the design and implementation are well received even though user involvement, in spite of being mentioned as indispensable for a Goal-Directed Task Analysis, was limited in this study. This study only had the ability to involve a single user once during the execution of the Goal-Directed Task Analysis (GDTA), and still yielded a good (at least for this study detailed and extensive enough) result. This shows that indirect sources can serve as an alternative to absorb the lack of user involvement.

Finally, the way that the user-tests were performed allowed the test subjects to participate from their home (which served as their working place during this study due to the measures concerning COVID-19). This way of testing was expected to yield less usable results as it was expected to be harder for the participants to understand the created prototype. This is not visible in the result however and can be useful for projects with limited user involvement like these. Enabling participants to participate online saves them traveling time to the test location and saves the test subject time. The fact that participation in an experiment takes the subject less than two hours instead of half a day could make it easier to find participants for these kinds of projects.

Remarks

A note has to be placed with respect to the effect of the added functionalities on SA. The MARS technique [26] was used to measure the expected effect on SA. Although this technique is developed as a subjective self-assessment device for training scenarios, the user-test performed in this project did not concern a self-assessment. The test subjects were not performing the hasty attack themselves but had to observe how the mission was executed. However, However, the MARS is still used because it provides a subjective measurement that is suited for training scenarios. The test subjects have experience as a PC and it is therefore believed that they can empathize with the PC in the virtual world and make a good estimation of how the functionalities affect the SA of the PC.

The usefulness of the composite methodology is questioned the most by the feedback on the improvised explosive device (IED) predicting functionality. This functionality received the most criticism from both test subjects. It could be questioned whether this result shows some deficiencies in the composite methodology as it is one of the results of the method.

However, the contrary should be considered as well. The IED functionality originated from a suggestion from the PC in chapter 5 to have a prediction of enemy elements. However, the target user does not have knowledge of human factors or AI, and therefore the designer will always need to find a balance between a suggestion from a target user and a possible implementation. In this specific case the result was not received positively. On the other hand, the current project also contains a lot of elements that were not even mentioned by the PC during the evaluation phase, but were received very well, like the auditory messages and speech commands for example. The point here is that the composite methodology is aimed at quick alterations and experiments. This focus on evaluations and iterations, will quickly identify these mistakes. It should therefore be argued that the identified issues with the IED functionality show one of the strengths of the composite methodology: identifying issues in an early stage of development. If time is directly spent on designing and developing a functional product, and the criticism about the IED functionality was found at the end of this process, a lot of work would have been wasted.

9.2.2 Artificial intelligence and Human Machine Cooperation

This project used (and simulated) two different kinds of AI. Some of these algorithms were actually implemented (monitoring friendly fire, and monitoring the movement of the sections) while others where simulated (IED prediction, eye tracker, and the conversational agent). The IED predicting functionality simulated a machine learning algorithm to come up with it's predictions, while the functionalities that check for sections diverting and section causing a risk of fratricide used rule-based AI. The reviews of the functionalities were mixed: the subjects were more reluctant to the functionality that uses a machine learning algorithm, but were very positive about the functionalities that used rule based algorithms. Both subjects immediately had their concerns with how realistic the IED prediction were, but didn't doubt the functionalities that monitors the sections movement and risks for friendly fire

This result gets more interesting when this is compared with the earlier treated literature that indicates the tendency of humans to be somewhat reticent in believing and using the outcomes of automated technologies [10]. The results of this study suggest that a nuance is needed in this statement, as the automated rule-based functionalities were not at all received reluctantly. Of course, the results are only based on two subjects, but there might be a sound explanation for this difference in acceptance of automated functionalities between rule-based algorithms and machine learning based algorithms.

The difference between a rule based algorithm and a machine learning algorithm is that a rule based system is by definition easier for a human to understand. The entire idea behind machine learning algorithms is that they can uncover and detect patterns and trends in data that are not directly evident to humans [41]. Rule based systems follow a defined set of rules, and all the conditions concerning these rules can be checked by the PC, making it easier to validate the information produced by the algorithm. The fact that machine learning algorithms produce information that a human can mostly not produce in the first place, prevents the PC from validating these results. In essence you could say that it is harder for a PC to detect flaws in machine learning algorithms, compared to flaws in a rule based system. This could explain why it might be easier for a PC to develop trust in a rule-based system in comparison to a machine learning based system, considering the fact that trustworthiness in AI is expected to reinforce and facilitate the collaboration between humans and AI [10].

The fact that incorrect classifications or predictions are more difficult to detect in machine learning based algorithms is double as important when it is considered that due to its statistical nature these machine learning algorithms always have a certain accuracy level, and therefore will also inevitably make mistakes. Since an IED predicted on a location where none turns out to be (false positive) bears less direct consequences than an IED on a location where none was predicted (false negative), these algorithms might be tweaked in order to sacrifice some accuracy in order to specifically limit the amount of false negatives, but are very unlike to prevent entirely, especially for complex predictions like IED locations. This combination of the difficulty to detect mistakes and at the same time the need to expect mistakes is a toxic combination in a situation where time is limited, stress levels are high, and where mistakes can lead to the loss of life.

To discuss this topic even further, another fact may also play a role in this difference in this particular project. When the monitoring (rule-based) functionalities are considered, it can be seen that the way that they are presented to the PC is actually explanatory itself. Consider a section that diverts from its route for example. When a PC receives a message that a section is diverting, the Raptor directly explains why it thinks that this section is diverting: it draws the location of the section relative to the route and the red line explains how the Raptor calculated the required extra time to get back to the route. In its visual presentation the rule-based algorithm explains how it ended up with its information and allows the PC to check this. When the IED predicting (and machine learning based) functionality is considered, its presentation is far less explanatory. This functionality just shows where it expects an IED and how certain it is of this prediction. It doesn't show factors that played an important role in creating this prediction, like the fact that a location forms an important approach route or cross road. This shows that explanations of AI can not only be conveyed in a textual form, like presented by Garcia and colleagues [10], but that they can also be conveyed visually. This is an important finding, since the common way of explaining answers of AI step by step and in textual form is not suitable for situations like a hasty attack where time is limited, stress levels are high and the consequences of mistakes can lead to the loss of life [15].

This preference of the PC for rule-based, or at least verifiable, algorithms did not only become visible in the reluctance towards the IED predicting functionality. The suggestions that were made during the user-test to widen the prediction of enemy elements beyond IEDs by including strategic locations on a map show this as well. While explaining what he meant with strategic locations, the subject described them very deterministic and based on a set of features. This way of describing these locations corresponds greatly to the characteristics of a rule-based system. It means that a location on a map is considered important when it meets a set of pre-defined requirements. This pull towards these rule-based systems might be explained by their verifiability. It could be argued that the fact that rule-based systems are easily explainable makes them better suitable to be used in environments that suffer a lot from reluctance against outcomes of automated systems. Machine learning based algorithms might be harder to explain and therefore to trust, which poses additional design challenges in the form of e.g. visualisation of the reasoning process of the algorithm.

Another issue with AI is that it can be complex to realize in a final project. This fact might be daunting at first sight and discourage designers to use these techniques in their systems. However, one of the experiences of this project is that AI can be simulated very well. The implication of this is that it becomes easy to transcend this complexity and use the strengths that this technology brings with it. The best thing about this combination is the fact that easy simulatable technology also allows for quick iterations. That means that an envisioned technology can be quickly transformed into a very effective solution even before a lot of programming has been done.

Therefore this project advises once more to simulate the use of AI technologies during the design and development phase of a project that wishes to use these technologies. The extent of this advice already became apparent in chapter 7, where the question was raised if the evaluation artifact could be kept more simple (e.g. less responsive). This notion was confirmed in chapter 8, where the user-test showed that the future fratricide prediction functionality that was explained using a mock-up could be equally well tested as the implemented functionalities that were turned into a simulated but responsive prototype. Although it is already a common practice to simulate digital solutions before they are developed, this once more presses on the importance to test as simple as possible. The benefits of this will be twofold: First of all it will allow for quicker iterations and faster progress of the design. Secondly it can save a lot of money, as it is considered common knowledge that identifying issues early in the design and development process saves massive amounts of money and time in comparison to when the same issues would have been identified at the end stages of development.

9.2.3 Decision making and human factors

One of the most promising results concerning decision making were the remarks of the two subjects that the added functionalities did not only help to increase higher-order SA levels, but at the same time are expected to lower the cognitive workload that is needed to obtain these levels of SA. This suggests that the created solution will support the PC in his decision making on two fronts at the same time: both an increase in SA [1] as well as a reduction in cognitive load [18] are considered to have a positive effect on decision-making. Finally the presented functionalities increased the decision-making performance of the PC even more by protecting the PC to the risks of temporal narrowing; one of the known human strategies to deal with high cognitive load, that can lead to the PC missing mission critical cues like the occurrence of fratricide [19] [18].

Of course, based on the performed user-test and the limited subjects, the conclusion can not be drawn that the added functionalities indeed increase the higher-order SA levels and decrease the cognitive load, but it does show reasons to further investigate this. The mixed reality environment in which the handheld prototype has been developed offers the ability to researchers to investigate these hypotheses under circumstances mimicking a real-life scenario [4]. At the same time, recent developments in measuring cognitive load continuously and during physical activities could allow us to investigate whether the solution indeed decreases the cognitive load or not [42] [43].

One of the possible reasons why the developed functionalities might indeed decrease the cognitive load of the PC is that they adhere to the mental models that the PC normally uses to cognitively compose level 2 SA. The monitoring functionalities improve level 2 SA by interpreting and combining different information sources, which is exactly how Endsley describes that humans build level 2 SA themselves [1]. It sounds plausible that the monitoring functionalities automate a translation step of the available information, that saves the PC cognitive processing and therefore workload. Generating this kind of information is relatively easy and is very well suitable to do with a rule-based automated system, as mentioned in section 9.2.2.

As trivial as it sounds to present information this way, the current trend that military personnel need to analyze more data while the available time to analyze this information decreases [20] shows indications that there might still be room for improvement. As experienced in this research, the Goal-Directed Task Analysis performed in chapter 4 highly contributed to the understanding of which information the PC needs and why the PC needs this information. This analysis is therefore considered to be one of the reasons that the functionalities are expected to actually lower instead of increase the cognitive load needed to build SA.

9.3 Conclusion

This study presents three concepts of functionalities that can be added to Raptor and augment the SA of the PC. A combination of Socio-Cognitive Engineering, Designing for situation awareness, and heuristic evaluations have been used to identify, implement and evaluate SA requirements in order to improve the SA of the PC. No hard conclusions can be drawn because of the limited amount of test subjects, but the positive responses of the experienced PCs indicate that the method used in this study is very well usable to find ways to leverage the strengths of AI to add higher-order SA components to an existing system. For that reason it is likely that the method as presented and used in this study forms an answer to the main research question of this study: *"How can the current version of the Raptor be enhanced with higher-order SA components to increase the SA of the PC during a hasty attack?"* With input from an experienced PC three SA supporting functionalities were selected and implemented in a prototype to test the concepts:

1. **Monitoring whether sections divert from their planned route:** A rule-based AI is used to monitor the movement of each section and alerts the PC either visual or auditory when a section diverts too much from its route.

2. **Monitor risks of friendly fire:** A rule-based system monitors the location, orientation and movement of all the sections and identifies and predicts when a risk of friendly fire occurs. In that case the PC will be alerted either auditory or visually.

3. **Predicting and monitoring location on which an enemy has placed improvised explosive devices (IEDs):** A machine learning based AI is used to predict the most likely places for an enemy to place IEDs and a rule-based system is used to check whether sections are approaching these locations. The PC can take these locations in consideration during the formulation of a plan of attack and is notified visually or auditory when a section approaches the location of a suspected IED.

The user-tests show signs that the functionalities increase the higher SA levels, and at the same time make it easier for the PC to build these SA levels. The predicted risks of friendly fire and detection of sections that are diverting from their route were perceived as valuable for the PC. It is important for the PC to correctly understand the situation and detect cues that may have critical consequences for the execution of the mission. The IED predicting functionality received mixed feedback. The experienced PCs questioned how realistic these predictions were and were worried about a false sense of SA that it might create for the PC, making the PC oblivious for IEDs on other locations. Both subjects told that they already witnessed this effect with man made predictions and at the same questioned how realistic these machine made predictions would be. Both subjects suggested extending this functionality to predict key locations on a map to make it more useful, since this information has broader implication than IEDs. These locations could be a high building that serve a vantage point or a bridge that forms an important logistic connection. Information like this will help a PC in planning or re-planning an attack.

This study presents two ways of presenting information to the PC and introduces a new interaction possibility with the Raptor. It was noticed that the PC is not always looking at the screen of the Raptor. This study suggests the use of an eye-tracker to monitor whether the PC is looking at the Raptor or not and uses a conversational AI to send voice messages to the PC to make sure that PC does not miss mission critical cues when the PC is not using the Raptor. The PC can also use this conversational AI to control the functionalities and turn each individual alert on or off. These audio notifications were positively received, be it with the annotation that they need to follow the extra short communication protocol due to the urgency of their message and to prevent an overload on the radio channel.

This project showed that AI can be used to incorporate higher-order SA requirements in the Raptor. The results of the user-test suggest that the inclusion of these higher-order SA requirements increased the SA of the PC and reduced the cognitive load necessary to build this level of SA. However, future studies should verify that this is indeed the case with more extensive user-tests. For these studies it will be beneficial that the current prototype is created within a mixed reality environment that can be used to let a subject execute a hasty attack in a real-life-mimicking setting.

9.4 Future work

This project ended with the user-test of the envisioned technologies and presented a composite method in order to create higher-order SA components. The question remains what needs to be done with the results. In the first place, the results of the user-tests show enough reasons to continue the development of the envisioned functionalities, and it offers multiple ways to improve them. On the other hand, the question remains in which other ways AI can improve the SA of the PC.

The next step in developing the functionalities would be to create a new interaction. The IED prediction functionality needs to be reconsidered. A solution needs to be found for the risk of providing a false sense of SA or the functionality needs to be discarded. It is important for the continuation of the design and development process, that the prototype needs to be specified more and more. This level of specification will help in realizing the final product. The details of the design and its implementation could be developed further. Elements of the design that could not have been tested in this iteration should be included and the prototype should slowly be prepared for a fully interactive evaluation.

At the same time, the question remains whether AI could be used to improve the SA of the PC in other ways. The most likely answer is yes. The possible ways to implement AI are very extensive, so it will be likely that other possibilities exist equally well. This project specifically chose to incorporate AI to increase the higher levels of SA. It made sense to do so, not only because of the good combination between the strengths of AI and the kind of information of which higher-level SA components exist, but also because level 3 SA has been found to be more predictive for human decision-making performance [40].

A more specific question would therefore be to see if AI could increase level 1 SA as well. Research namely shows that most decision-making mistakes originate from level 1 SA requirements that are not noticed. However, the difficulty in this issue is which role AI would play. According to Endsley [1], missing level 1 SA happens due to a combination of multiple psychological factors like the capacity of the working memory, stress, and attention that are all influencing each other. A possible solution to make the PC miss less level 1 requirement would be to let AI help in directing the attention of the PC. The drawback of this solution is that it takes away some autonomy of the PC and lets the AI decide what is important in that situation and what not. Besides the fact that this is very difficult to do, it is also strongly discouraged. Previous research shows signs that this will actually have a bad effect on the SA of the PC [7].

Another way to let AI support the SA of the PC was suggested during a user-test. Perhaps AI can be used to interpret the so-called key areas in an environment. This information would help the PC to better understand the current situation and therefore would increase the second level of SA of the PC a lot. Doing so, it follows the same principle as earlier in this project: adding higher-order SA requirements. Besides the functionalities that are presented in this project, there probably exist more possible higher-order SA requirements that still can be added to the Raptor. Perhaps an elaboration on the GDTA will identify these. However, all these functionalities will adhere to the same principle, which in my personal opinion still seems the most logical way to use AI to increase SA: compute higher-order SA requirements.

Bibliography

[1] M. Endsley, “Endsley, m.r.: Toward a theory of situation awareness in dynamic systems. human factors journal 37(1), 32-64,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 37, pp. 32–64, 03 1995.

[2] “Sluwe voss,” Feb 2019. [Online]. Available: https://magazines.defensie.nl/materieelgezien/2019/01/10_sluwe_voss

[3] “Het eerste voss in het vizier,” Jan 2018. [Online]. Available: https://vovklict.nl/intercom/2018/1/17.pdf

[4] M. van Beurden and L. Roijendijk, “Towards a mixed reality platform for applied cognitive load evaluation,” in *Engineering Psychology and Cognitive Ergonomics*. Springer International Publishing, 2019, pp. 123–136. [Online]. Available: https://doi.org/10.1007/978-3-030-22507-0_10

[5] M. Endsley, “A survey of situation awareness requirements in air-to-air combat fighters,” *International Journal of Aviation Psychology - INT J AVIAT PSYCHOL*, vol. 3, pp. 157–168, 04 1993.

[6] M. A. Neerincx, “Situated cognitive engineering for crew support in space,” *Personal and Ubiquitous Computing*, vol. 15, no. 5, pp. 445–456, Jul. 2010. [Online]. Available: https://doi.org/10.1007/s00779-010-0319-3

[7] M. Endsley, *Designing for situation awareness: an approach to user-centered design*. Boca Raton, FL: CRC Press, 2012.

[8] J. Nielsen and R. Molich, “Heuristic evaluation of user interfaces,” in *Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI ’90*. ACM Press, 1990. [Online]. Available: https://doi.org/10.1145/97243.97281

[9] M. Sutrop, “SHOULD WE TRUST ARTIFICIAL INTELLIGENCE?” *Trames. Journal of the Humanities and Social Sciences*, vol. 23, no. 4, p. 499, 2019. [Online]. Available: https://doi.org/10.3176/tr.2019.4.07

[10] I. Garcia-Magarino, R. Muttukrishnan, and J. Lloret, “Human-centric AI for trustworthy IoT systems with explainable multilayer perceptrons,” *IEEE Access*, vol. 7, pp. 125 562–125 574, 2019. [Online]. Available: https://doi.org/10.1109/access.2019.2937521

[11] M. A. Neerincx, J. van der Waa, F. Kaptein, and J. van Diggelen, “Using perceptual and cognitive explanations for enhanced human-agent team performance,” in *Engineering Psychology and Cognitive Ergonomics*. Springer International Publishing, 2018, pp. 204–214. [Online]. Available: https://doi.org/10.1007/978-3-319-91122-9_18

[12] C. S. de Oliveira, F. Giustozzi, C. Zanni-Merk, C. Sanin, and E. Szczerbicki, “Stream reasoning to improve decision-making in cognitive systems,” *Cybernetics and Systems*, vol. 51, no. 2, pp. 214–231, Jan. 2020. [Online]. Available: https://doi.org/10.1080/01969722.2019.1705553

[13] P. van Maanen, J. Lindenberg, and M. Neerincx, “Integrating human factors and artificial intelligence in the development of human-machine cooperation,” in *Proceedings of the 2005 International Conference on Artificial Intelligence (ICAI’05)*, H. Arabnia and R. Joshua, Eds. CSREA Press, 2005, pp. 10–16, sPIE.

[14] D. Engelbart, “Augmenting human intellect: A conceptual framework,” 1962.

[15] M. A. Razzak and M. N. Islam, “Exploring and evaluating the usability factors for military application: A road map for HCI in military applications,” *Human Factors and Mechanical Engineering for Defense and Safety*, vol. 4, no. 1, Jan. 2020. [Online]. Available: https://doi.org/10.1007/s41314-019-0032-6

[16] B. C. Dostal, “Enhancing situational understanding through the employment of unmanned aerial vehicles,” *Interim Brigade Combat Team Newsletter*, Jan 2007.

[17] G. Klein, B. Moon, and R. Hoffman, “Making sense of sensemaking 1: Alternative perspectives,” *IEEE Intelligent Systems*, vol. 21, no. 4, pp. 70–73, Jul. 2006. [Online]. Available: https://doi.org/10.1109/mis.2006.75

[18] I. Cohen, W.-P. Brinkman, and M. A. Neerincx, “Modelling environmental and cognitive factors to predict performance in a stressful training scenario on a naval ship simulator,” *Cognition, Technology & Work*, vol. 17, no. 4, pp. 503–519, Feb. 2015. [Online]. Available: https://doi.org/10.1007/s10111-015-0325-3

[19] H. de Greef and M. Neerincx, "Cognitive support: designing aiding to supplement human knowledge," *International Journal of Human-Computer Studies*, vol. 42, no. 5, pp. 531–571, May 1995. [Online]. Available: https://doi.org/10.1006/ijhc.1995.1023

[20] S. Rahimi, H. Mohamed, and N. Paredes, "Company command support system CSˆ2," in *2007 IEEE Systems and Information Engineering Design Symposium*. IEEE, Apr. 2007. [Online]. Available: https://doi.org/10.1109/sieds.2007.4374016

[21] R. Kumar, M. Aggarwal, T. E. Germano, T. Zhao, R. Fontana, and M. Bushika, "Tag-n-track system for situation awareness for MOUTs," in *Enabling Technologies for Simulation Science X*, D. A. Trevisani, Ed. SPIE, May 2006. [Online]. Available: https://doi.org/10.1117/12.665846

[22] C. M. Jonker, M. B. van Riemsdijk, and B. Vermeulen, "Shared mental models," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2011, pp. 132–151. [Online]. Available: https://doi.org/10.1007/978-3-642-21268-0_8

[23] M. Johnson, J. M. Bradshaw, P. J. Feltovich, C. M. Jonker, M. B. V. Riemsdijk, and M. Sierhuis, "Coactive design: Designing support for interdependence in joint activity," *Journal of Human-Robot Interaction*, vol. 3, no. 1, p. 43, Mar. 2014. [Online]. Available: https://doi.org/10.5898/jhri.3.1.johnson

[24] J. Nielsen, "Finding usability problems through heuristic evaluation," in *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '92*. ACM Press, 1992. [Online]. Available: https://doi.org/10.1145/142750.142834

[25] M. Endsley, D. Garland, R. Wampler, and M. Matthews, "Modeling and measuring situation awareness in the infantry operational environment," p. 126, 01 2000.

[26] M. Matthews and S. Beal, "Assessing situation awareness in field training exercises," *U.S. Army Research Institute for the Behavioral and Social Sciences*, p. 31, 09 2002.

[27] M. Duistermaat, C. C. MPhil, and S. de Vries, "Informatiebehoefte van de uitgestegen militair en zijn groep," *TNO Defensie en Veiligheid, Internal Document*, June 2009.

[28] N. Smets, J. Streefkerk, and K. Kranenborg, "Commander's dashboard - future decision support for dismounted group commanders," *TNO Defensie en Veiligheid, Internal Document*, March 2010.

[29] M. V. Beurden, L. Roijendijk, and S. van Trip, “Cognitive workload during a hasty attack by the royal netherlands marine corps,” *TNO Defensie en Veiligheid, Internal Document*, june 2018.

[30] “Max velocity tactical.” [Online]. Available: https://www.youtube.com/user/maxvelocitytactical

[31] J. G. Hollands, T. Spivak, and E. W. Kramkowski, “Cognitive load and situation awareness for soldiers: Effects of message presentation rate and sensory modality,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 61, no. 5, pp. 763–773, Feb. 2019. [Online]. Available: https://doi.org/10.1177/0018720819825803

[32] Y. M. Fat, Y. Weiming, and Y. K. Leng, “Analysis of weapon danger area on fire support for flanking troops,” *DSTA Horizons*, 2017.

[33] R. W. Hill, Y. Kim, and J. Gratch, “Anticipating where to look: Predicting the movements of mobile agents in complex terrain,” ser. AAMAS ’02. Association for Computing Machinery, 2002, p. 821–827. [Online]. Available: https://doi.org/10.1145/544862.544935

[34] “Sce example home,” https://confluence.ewi.tudelft.nl/display/SE/SCE+Example+Home, accessed: 2020-12-24.

[35] “App-6(c) nato joint militairy symbology,” may 201w.

[36] [Online]. Available: https://multisim.nl/

[37] H. Vreij, “Dutch mod launches investigation into ’friendly fire’ incidents,” 01 2008.

[38] K. Ridler, K. Gannon, S. Peoples, G. Ziezulewicz, and L. S. I. Gould, “Report: B-1b bombs killed u.s. soldiers after location mix-up,” Aug 2017. [Online]. Available: https://www.militarytimes.com/2014/09/05/report-b-1b-bombs-killed-u-s-soldiers-after-location-mix-up/

[39] C. Wickens, *Engineering psychology and human performance*. Boston: Pearson, 2013.

[40] M. R. Endsley, “Final reflections,” *Journal of Cognitive Engineering and Decision Making*, vol. 9, no. 1, pp. 101–111, Feb. 2015. [Online]. Available: https://doi.org/10.1177/1555343415573911

[41] P. Boldi and C. Monti, “Llamafur: Learning latent category matrix to find unexpected relations in wikipedia,” in *Proceedings of the 8th ACM Conference on Web Science*, ser. WebSci ’16. New York, NY, USA: Association for Computing Machinery, 2016, p. 218–222. [Online]. Available: https://doi.org/10.1145/2908131.2908153

[42] K. Gramann, D. P. Ferris, J. Gwin, and S. Makeig, “Imaging natural cognition in action,” *International Journal of Psychophysiology*, vol. 91, no. 1, pp. 22–29, Jan. 2014. [Online]. Available: https://doi.org/10.1016/j.ijpsycho.2013.09.003

[43] S. Debener, F. Minow, R. Emkes, K. Gandras, and M. de Vos, “How about taking a low-cost, small, and wireless EEG for a walk?” *Psychophysiology*, vol. 49, no. 11, pp. 1617–1621, Sep. 2012. [Online]. Available: https://doi.org/10.1111/j.1469-8986.2012.01471.x

TNO innovation for life

UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering, Mathematics & Computer Science

Appendices

Adding higher-order Situation Awareness components to a Platoon Commander's Battle Management System.

Daan H. Ekkelenkamp

January 2021

Appendix A: List with all the D4SA design principles

General design principles

1. Organize information around goals
2. Present Level 2 information directly - support comprehension
3. Provide assistance for Level 3 projections
4. Support global SA
5. Support trade-offs between goal-driven and data-driven processing
6. Make critical cues for schema activation salient
7. Take advantage of parallel processing capabilities
8. Use information filtering carefully

Certainty design principles

9. Explicitly identify mission information
10. Support sensor reliability assessment
11. Use data salience in support of certainty
12. Represent information timeliness
13. Support assessment of confidence in composite data
14. Support uncertainty management activities

Complexity design Principles

15. Just say no to to feature creep - buck the trend
16. Manage rampant featurism through prioritization and flexibility
17. Insure logical consistency across modes and features
18. Minimize logic branches
19. Map system functions to the goal and mental model of users
20. Provide system transparency and observability
21. Group information based on Level 2/3 SA requirements and goals
22. Reduce display density, but don't sacrifice coherence
23. Provide consistency and standardization on controls across different displays and systems
24. Minimize task complexity

Alarm design principles

25. Don't make people rely on alarms - provide projection support
26. Support alarm confirmation activities
27. Make alarms unambiguous
28. Reduce false alarms, reduce false alarms, reduce false alarms
29. Set missed alarm and false alarm trade-offs appropriately
30. Use multiple modalities to alarm but insure they are consistent

31. Minimize alarm disruptions to ongoing activities
32. Support the assessment and diagnosis of multiple alarms
33. Support the rapid development of global SA of systems in an alarm state

Automation design principles

34. Automate only if necessary
35. Use automation for assistance in carrying out routine actions rather than higher level cognitive tasks
36. Provide SA support rather than decisions
37. Keep the operator in control and in the loop
38. Avoid the proliferation of automation modes
39. Make modes and system states salient
40. Enforce automation consistency
41. Avoid advanced queuing of tasks
42. Avoid the use of information queuing
43. Use methods of decision support that create human/system symbiosis
44. Provide automation transparency

Multioperator design principles

45. Build a common picture to support team operations
46. Avoid display overload in shared displays
47. Provide flexibility to support shared SA across functions
48. Support transmission to different comprehension and projections across teams
49. Limit nonstandardization of display coding techniques
50. Support transmission of SA within positions by making status of elements and states overt

Appendix B: Nielsen's nine heuristics

Simple and Natural Dialogue
Dialogues should not contain irrelevant or rarely needed information. Every extraneous unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. All information should appear in a natural and logical order.

Speak the User's Language
The dialogue should be expressed clearly in words, phrases, and concepts familiar to the user rather than in system-oriented terms.

Minimize the User's Memory Load.
The user's short-term memory is limited. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate. Complicated instructions should be
simplified.

Be Consistent
Users should not have to wonder whether different words, situations, or actions mean the same thing. A particular system action- when appropriate-should always be achievable by one particular user action. Consistency also means coordination between subsystems and between major independent systems with common user populations [7].

Provide Feedback
The system should always keep the user informed about what is going on by providing him or her with appropriate feedback within reasonable time.

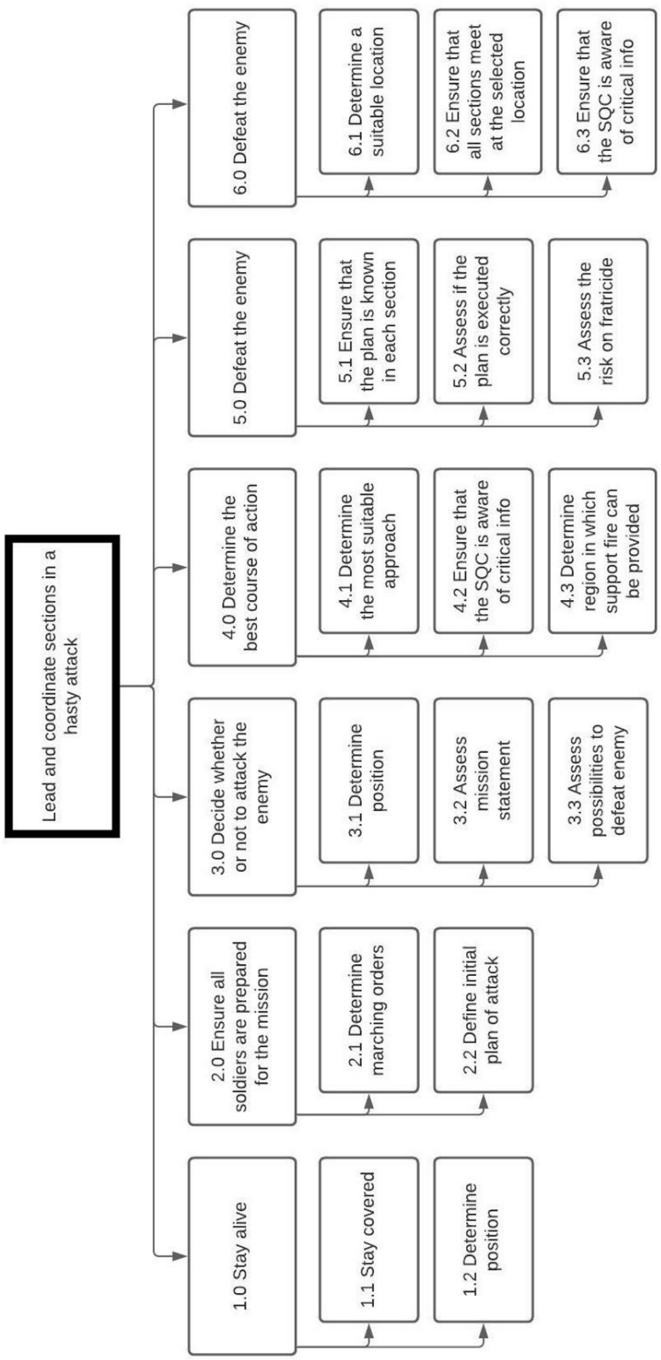
Provide Clearly Marked Exits
A system should never capture users in situations that have no visible escape. Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue.

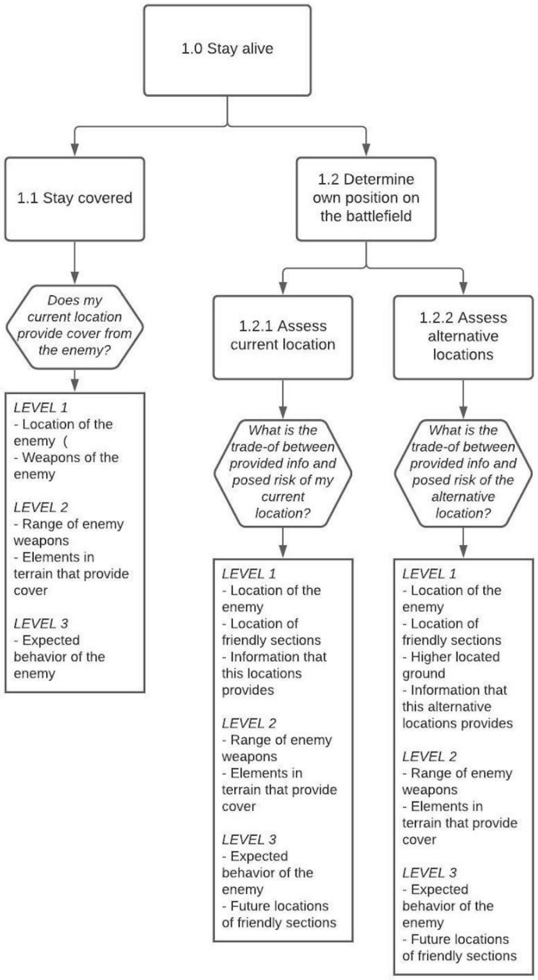
Provide Shortcuts
The features that make a system easy to learn-such as verbose dialogues and few entry fields on each display are often cumbersome to the experienced user. Clever shortcuts unseen by the novice user-may often be included in a system such that the system caters to both inexperienced and experienced users.

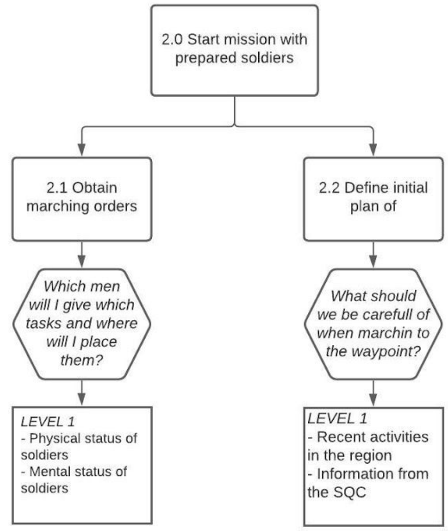
Provide Good Error Messages
Good error messages are defensive, precise, and constructive. Defensive error messages blame the problem on system deficiencies and never criticize the user. Precise error messages provide the user with exact information about the cause of the problem. Constructive error messages provide meaningful suggestions to the user about what to do next.

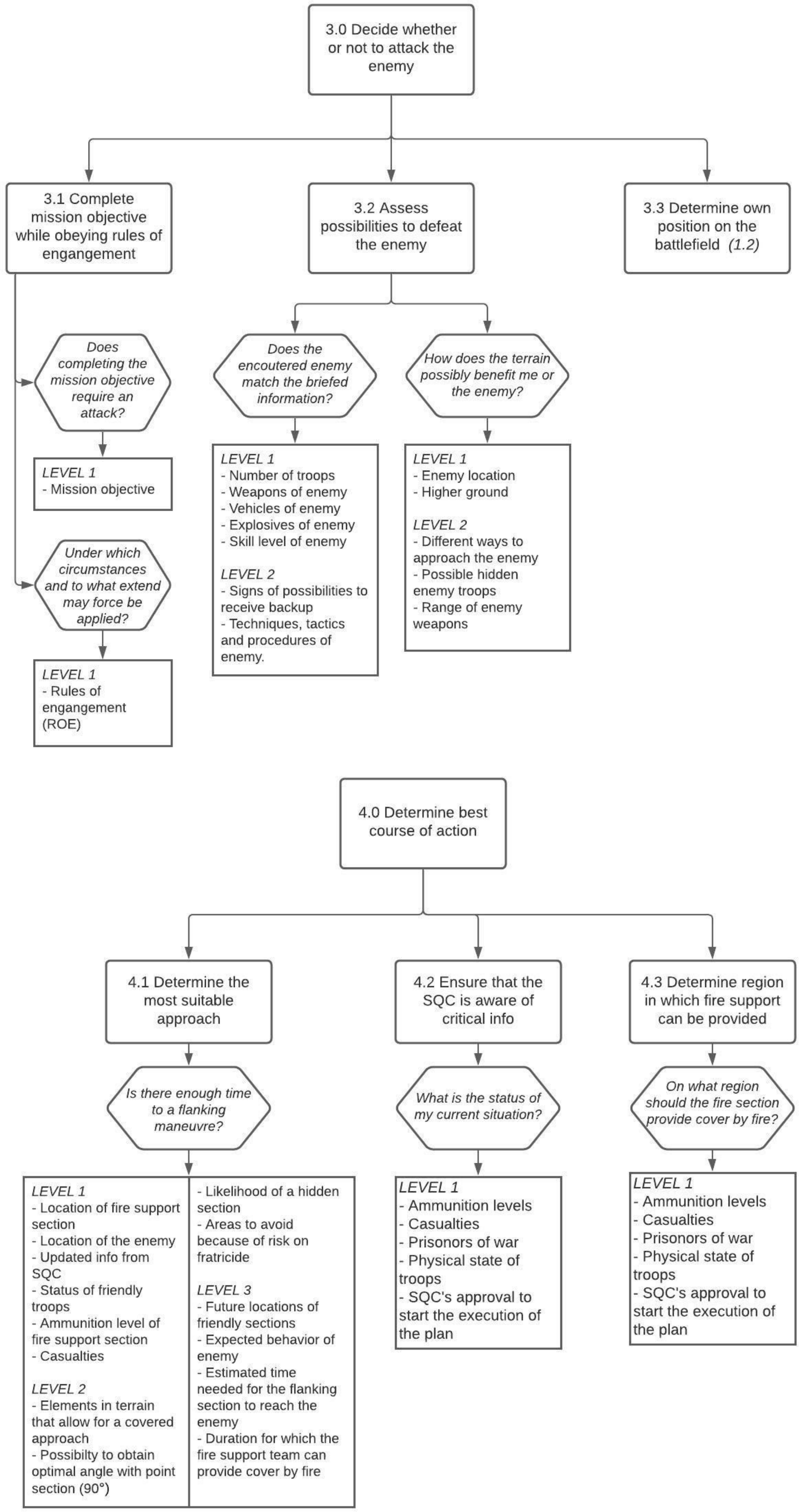
Error Prevention
Even better than good error messages is a careful design that prevents a problem from occurring in the first place.

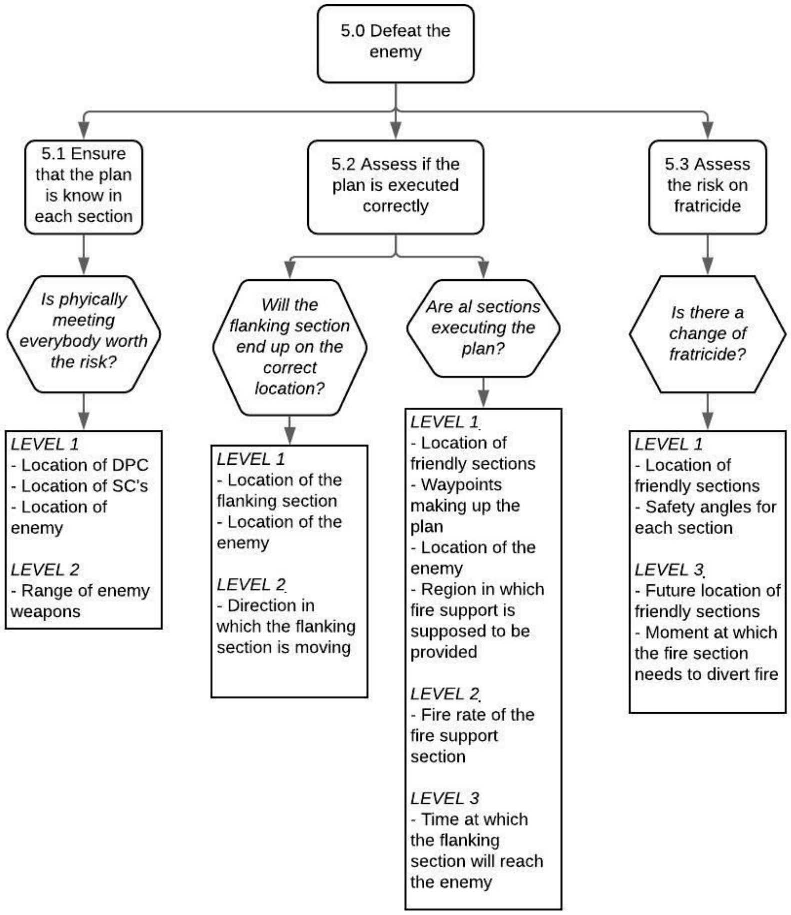
Appendix C: Complete goal structure of the GDTA

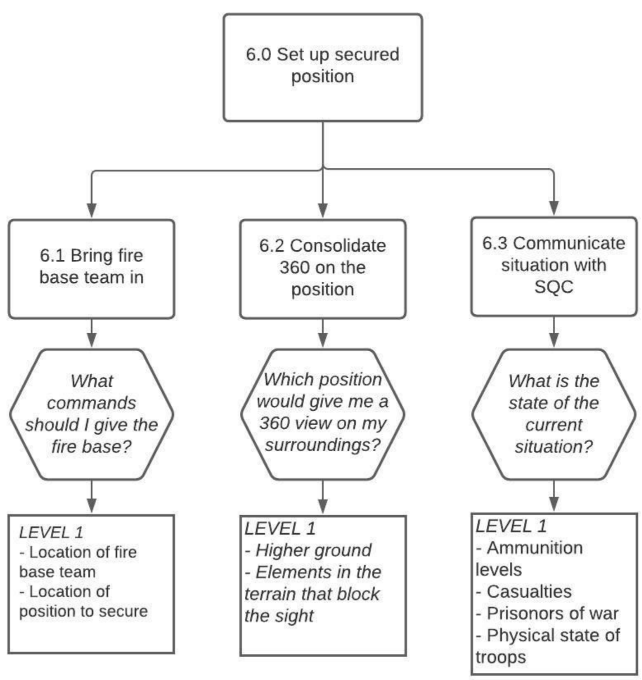



1.0 Stay alive
1.1 Stay covered
1.2 Determine own position on the battlefield
Does my current location provide cover from the enemy?
LEVEL 1
- Location of the enemy (
- Weapons of the enemy
LEVEL 2
- Range of enemy weapons
- Elements in terrain that provide cover
LEVEL 3
- Expected behavior of the enemy
1.2.1 Assess current location
1.2.2 Assess alternative locations
What is the trade-of between provided info and posed risk of my current location?
What is the trade-of between provided info and posed risk of the alternative location?
LEVEL 1
- Location of the enemy
- Location of friendly sections
- Information that this locations provides
LEVEL 2
- Range of enemy weapons
- Elements in terrain that provide cover
LEVEL 3
- Expected behavior of the enemy
- Future locations of friendly sections
LEVEL 1
- Location of the enemy
- Location of friendly sections
- Higher located ground
- Information that this alternative locations provides
LEVEL 2
- Range of enemy weapons
- Elements in terrain that provide cover
LEVEL 3
- Expected behavior of the enemy
- Future locations of friendly sections


2.0 Start mission with prepared soldiers
2.1 Obtain marching orders
2.2 Define initial plan of
Which men will I give which tasks and where will I place them?
What should we be carefull of when marchin to the waypoint?
LEVEL 1
- Physical status of soldiers
- Mental status of soldiers
LEVEL 1
- Recent activities in the region
- Information from the SQC


3.0 Decide whether or not to attack the enemy
3.1 Complete mission objective while obeying rules of engangement
3.2 Assess possibilities to defeat the enemy
3.3 Determine own position on the battlefield (1.2)
Does completing the mission objective require an attack?
LEVEL 1
- Mission objective
Under which circumstances and to what extend may force be applied?
LEVEL 1
- Rules of engangement (ROE)
Does the encoutered enemy match the briefed information?
LEVEL 1
- Number of troops
- Weapons of enemy
- Vehicles of enemy
- Explosives of enemy
- Skill level of enemy
LEVEL 2
- Signs of possibilities to receive backup
- Techniques, tactics and procedures of enemy.
How does the terrain possibly benefit me or the enemy?
LEVEL 1
- Enemy location
- Higher ground
LEVEL 2
- Different ways to approach the enemy
- Possible hidden enemy troops
- Range of enemy weapons
4.0 Determine best course of action
4.1 Determine the most suitable approach
4.2 Ensure that the SQC is aware of critical info
4.3 Determine region in which fire support can be provided
Is there enough time to a flanking maneuvre?
LEVEL 1
- Location of fire support section
- Location of the enemy
- Updated info from SQC
- Status of friendly troops
- Ammunition level of fire support section
- Casualties
LEVEL 2
- Elements in terrain that allow for a covered approach
- Possibilty to obtain optimal angle with point section (90°)
- Likelihood of a hidden section
- Areas to avoid because of risk on fratricide
LEVEL 3
- Future locations of friendly sections
- Expected behavior of enemy
- Estimated time needed for the flanking section to reach the enemy
- Duration for which the fire support team can provide cover by fire
What is the status of my current situation?
LEVEL 1
- Ammunition levels
- Casualties
- Prisonors of war
- Physical state of troops
- SQC's approval to start the execution of the plan
On what region should the fire section provide cover by fire?
LEVEL 1
- Ammunition levels
- Casualties
- Prisonors of war
- Physical state of troops
- SQC's approval to start the execution of the plan


5.0 Defeat the enemy
5.1 Ensure that the plan is know in each section
5.2 Assess if the plan is executed correctly
5.3 Assess the risk on fratricide
Is phyically meeting everybody worth the risk?
Will the flanking section end up on the correct location?
Are al sections executing the plan?
Is there a change of fratricide?
LEVEL 1
- Location of DPC
- Location of SC's
- Location of enemy
LEVEL 2
- Range of enemy weapons
LEVEL 1
- Location of the flanking section
- Location of the enemy
LEVEL 2
- Direction in which the flanking section is moving
LEVEL 1
- Location of friendly sections
- Waypoints making up the plan
- Location of the enemy
- Region in which fire support is supposed to be provided
LEVEL 2
- Fire rate of the fire support section
LEVEL 3
- Time at which the flanking section will reach the enemy
LEVEL 1
- Location of friendly sections
- Safety angles for each section
LEVEL 3
- Future location of friendly sections
- Moment at which the fire section needs to divert fire


6.0 Set up secured position
6.1 Bring fire base team in
6.2 Consolidate 360 on the position
6.3 Communicate situation with SQC
What commands should I give the fire base?
Which position would give me a 360 view on my surroundings?
What is the state of the current situation?
LEVEL 1
- Location of fire base team
- Location of position to secure
LEVEL 1
- Higher ground
- Elements in the terrain that block the sight
LEVEL 1
- Ammunition levels
- Casualties
- Prisonors of war
- Physical state of troops

Appendix D: Insights in the consideration of a PC during a hasty attack

The interview with the PC in which the preliminary goal structure was evaluated provided clear insights into some of the important considerations that a PC makes during the execution of a hasty attack.

All actions of the PC should be about leading and coordinating his sections
The first comment made by the former PC is that every PC has one major goal. This major goal is not “to stay alive” or to “make a plan of attack”, but to lead and coordinate his sections. With exceptions of drills (trained military procedures), the sections only perform actions when the PC has given them the command to do so. This means that everything the PC does should finally support the PC in leading and coordinating his sections. Staying alive and making a plan of attack is only part of this.

Positioning in the battlefield
During all phases of the hasty attack, the PC is constantly reconsidering his position. This consideration weighs the risk that his current position offers against the information that this location provides him. To make a good plan and coordinate his sections well, he needs to have access to the information components listed in the goal hierarchy like the terrain characteristics and location of the enemy. The PC can access the majority of this information when located on the frontline, but there is a high risk involved in staying at the frontline. Staying behind all the sections is safer, but the PC is not able to access a lot of information needed to lead and coordinate from this location. It is also not at all the case that the PC stays with a single section all the time. The PC can position himself in between sections for various reasons. First of all, fire tends to draw fire, increasing the risk that is involved with staying with a section, especially if that section is firing on the enemy. Furthermore, on contact, the section will perform its drills as an automated reaction. When the PC is staying with a section while they are performing their drills, the PC can get in their way. If available, higher ground will often offer a good option to position oneself. It favors the information that the PC can access and reduces the risk of getting shot.

Asses the enemy force
The platoon received information about the enemy during the briefing. What weapons do they tend to carry, with how many sections do they tend to move, do they have mortars? When contact is made with the enemy this known information is compared to what is encountered. The section under fire reports information about the enemy to the PC. If the PC notices something that differs from the briefing (the presence of mortars while these were believed to be absent for example) this is a reason to question the other information from the briefing. The known tactics, techniques, and procedures (TTP) of the enemy are also taken into account in

this phase, which can help in determining a way to deal with the enemy. If the information the PC received during the briefing is incorrect, then more time is needed to assess the actual force of the enemy they are encountering.

Determine to attack or not
The most important factor in determining whether or not to proceed to an attack is the goal of the mission. Some mission objectives are to locate enemy troops. When this is the goal of the mission, a PC will hardly ever put his platoon in the danger of attacking the enemy. Furthermore, the way in which the platoon will act also depends on the Rules Of Engagement (ROE). These rules define the circumstances, conditions, manners, and degree of force that might be applied.

Decide the approach direction for the attack
Making a flanking movement offers the best possibility to provide contact fire. However, it is not always possible to make a flanking movement. One example is a very close enemy. Another one is terrain that does not offer an option to make a flanking movement. Performing a flanking movement creates the danger of fratricide with another platoon and within sections. When there are casualties there might be no time to perform a flanking maneuver as the casualties need to be retrieved as quickly as possible. During the creation of a plan of attack, the PC is estimating how long the flanking movement will take and whether the required time is too much given the current situation. Both estimations are made based on experience and gut-feeling, not on a calculation.

One of the important elements in determining the available time is the level of ammunition of the section providing fire support. This level of ammunition determines how long they can provide cover by fire and therefore determines the time the flanking section has to reach the enemy. How long this section can provide suppression fire is requested by the PC from the SC of the particular section. Whenever a flanking maneuver is not possible, a frontal approach needs to be made.

Making sure that everybody is aware of the plan for the attack
There are three possible ways to make sure that everybody understands the plan that the PC has made for the attack. The first one is to physically meet all the commanders and the deputy PC. This enables the PC to check very well whether everybody understands the plan correctly and to look each other in the eyes before the attack is going to begin, but comes with a risk of bringing all commanders together in a single location. The second option is to communicate the plan via the radio. The SCs should be able to understand and envision the plan good enough this way, that’s what they’re trained for. There is less risk involved, but it will be less clear to the PC whether everybody indeed correctly understood his plan. The third option is only possible with the new Raptor on which the PC can draw his plan for the SCs. This way the risk of bringing all the commanders together can be avoided while the plan should be very clear to everyone. The only downside is that the commanders cannot look each other in the eyes before the attack begins, something that most commanders tend to prefer.

Coordinating the attack

During the attack, the PC is constantly monitoring if all the sections are moving accordingly to the plan, and if there is no risk of fratricide between the sections. Often terrain that is difficult to see through is chosen for a flanking movement as this allows for a hidden approach. At the same time, this terrain is also difficult to move through. Therefore it often happens that sections have difficulties with reaching the intended location. A section may end up right in front of- or behind the enemy. The PC is constantly checking the location of the section concerning his plan. At the same time, he needs to make sure that there is no risk of friendly fire between his sections.

Appendix E: Use cases, requirements, and claims

UC001: A flanking section starts to diverge from the communicated path

Objective	OB01: Title
Actors	SH01: Platoon Commander (PC)
Pre-condition	PC communicated the plan using the Raptor and the plan includes a flanking movement through difficult terrain. The section that started making the flanking movement made some good progress.
Post-condition	The PC is notified and gives a corrective command which should get the section back on the plan.
Action sequence	1. The Raptor detects that the section's distance from the communicated path exceeds the threshold. 2. The Raptor checks if the PC is looking to the screen of the Raptor and detects that this is indeed the case 3. The Raptor gives a visual signal that the section moved too far away from the plan by highlighting the distance to the path and letting the icon of the concerning section glow. 4. The PC selects the section by pressing it. The previous locations of the section start to pop up. He sees from the previous locations of the section that they indeed slowly got off course and decides the give them correcting commands. 5. The Raptor picks up the commands from the PC and includes their current location in the plan and starts to monitor their current location again. Alternative steps 1.1 The Raptor detects that the section's distance from the communicated path exceeds the threshold. 1.2 The Raptor checks if the PC is looking to the screen of the Raptor and detects that this is not the case 1.3 The Raptor checks if there is communication via the radio, and waits 5 seconds for silence (of 1 second) before it starts to communicate his vocal notification: Raptor for Romeo (short pause): Section X if diverging from the plan. It might be useful to check their status with them. 1.4 The PC does not command the Raptor to stop monitoring the section and just gives the section the command to return to the path. Alternative steps: 2.1 The Raptor detects that the section's distance from the communicated path exceeds the threshold. 2.2 The Raptor checks if the PC is looking at the screen of the Raptor and detects that this is not the case. 2.3 The Raptor checks if there is communication via the radio, and waits 5 seconds for silence (of 1 second) before it starts to communicate his vocal notification: Raptor for Romeo (short pause): Section X if diverging from the plan. It might be useful to check their status with them. Let me know if you want me to stop the monitoring of section X. 2.4The PC does not command the Raptor to stop monitoring the section and just gives the section the command to return to the path. 2.5 The sections respond that they are not able to do so due to the terrain. A group of rocks is laying in their way. 2.6 The PC commands the Raptor to stop monitoring this section for a minute, as he thinks that the section will be back on the original path by that time. Romeo to Raptor: stop monitoring section X for 1 minute. 2.7 The Raptor stops monitoring the section for 1 minute and starts tracking it once the minute has passed.

UC002: A flanking section ends up behind the enemy line

Objective	OB01: Increase Situation Awareness of the Platoon Commander
Actors	SH01: Platoon Commander (PC)
Pre-condition	PC communicated the plan using the Raptor and the plan includes a flanking movement through difficult terrain.
Post-condition	The PC is notified and gives a corrective command which should improve the position of the section.
Action sequence	1. The Raptor detects that the flanking section has moved beyond the point at which they should turn towards the enemy. The Raptor informs the PC. When the raptor is active it gives this message visually. When the Raptor is not active it presents the message via a voice command: Raptor for Romeo: It seems as if section A has moved too far North and is now behind the enemy. 2. The PC contacts the section and tells it that they moved too far. The PC gives them some correcting commands. 3. The Raptor adds their current location to their path and keeps tracking them if they do reach the optimal spot this time. 4. The section stops at the optimal spot to approach the enemy and starts the attack. Alternative steps: 1.1 The Raptor detects that the flanking section has moved beyond the point at which they should turn towards the enemy. The Raptor informs the PC. When the raptor is active it gives this message visually. When the Raptor is not active it presents the message via a voice command: Raptor for Romeo: It seems as if section A has moved too far North and is now behind the enemy. Let me know if I need to stop tracking this section. 1.2 The PC contacts the sections and tells it that they have moved too far. 1.3 The section replies that the location looks slightly different than expected and that they temporarily need to travel slightly further North than to approach the enemy, but that they still expect to end up in a good location to approach the enemy. 1.4 The PC commands the Raptor to stop tracking the specific section and consults his raptor. 1.5 At the same time he informs the section that is providing support fire to be aware of the altered approach of the flanking section. 1.6 The Raptor informs the PC that the flanking section will walk into the fire of the section that provides support fire in 30 seconds. 1.7 The PC commands the support fire section to start diverting its fire. 1.8 The flanking section appears quite as a surprise and defeats the enemy force.

UC step	Requirements	Claims
1	RQ002: Detect a section behind enemy lines	CL002: The Raptor reassures that the PC will be aware of sections that end up behind enemy lines
1	RQ011: Detect when the PC is looking at the Raptor	CL007: The Raptor reassures that the PC does not receive redundant notifications

UC003: Acute risk of fratricide

Objective	OB01: Increase Situation Awareness of the Platoon Commander
Actors	SH01: Platoon Commander (PC)
Pre-condition	The Raptor detects an acute risk of fratricide.
Post-condition	The PC is notified and gives a corrective command which should improve the position of the section
Action sequence	1. Section A is approaching the enemy that section B is providing fire on. 2. The moment section A gets in the fire zone of section B, the Raptor immediately communicates this to the PC. Raptor to Romeo: Section A is having an acute risk on fratricide. Directly advised to let section B avert fire to the east. When PC is looking at the Raptor, also a very salient cue is immediately given. 3. The PC gives the command and section B indeed averts its fire. Luckily nobody from section A was shot. Alternative steps 1.1 Section A is making a flanking movement and approaching the enemy. 1.2 Section B was providing fire support but got relieved by section C, who has obtained a better firebase. 1.3 After a while section A is closing in on its way to the enemy location, not aware of the changed location from where support by fire is given. As a result, section A ends up in the line of fire of C earlier then expected. 1.4 The Raptor notices this and immediately contacts the PC. Raptor to Romeo: Section A is having an acute risk on fratricide. Directly advised to let section B avert fire to the east. When PC is looking at the Raptor, also a very salient cue is immediately given. 1.5 The PC gives the command and section B indeed averts its fire. Luckily nobodyfrom section A was shot. Alternative steps 2.1 Section A and B are both maneuvering towards the enemy. 2.2 Section B ran into an obstacle and needed to change its course. Five second ago the Raptor already warned the PC that section B diverted from the route and that A en B will end up in eachothers line of fire. 2.3 The PC commicated to section A and B to be extra cautious and expect contact with friendly troops 2.4 Indeed, some seconds later, A and B make contact with eachother. Because of the warning that was given by the PC to both the sections the Raptor does not give of an acute warning for fratricide, as both sections are already aware of this risk. Alternative steps 3.1 Section A and B are both maneuvering towards the enemy. 3.2 Section B ran into an obstacle and needed to change its course. Five second ago the Raptor already warned the PC that section B diverted from the route. 3.3 The changed course changes the location and direction from where section B will approach the enemy, and a building is now placed in between section A and section B. 3.4 The Raptor takes this building into account when detecting risks of fratricide and does not notify the PC.

UC004: Future risk of fratricide

Objective	OB01: Increase Situation Awareness of the Platoon Commander
Actors	SH01: Platoon Commander (PC)
Pre-condition	PC communicated the plan using the Raptor and the plan includes a flanking movement.
Post-condition	The PC is notified and gives a corrective command which should improve the position of the section.
Action sequence	1. Section A is making a flanking movement and starts to approach the enemy. 2. Sections B is providing support fire on the enemy in order to give A enough time to make the flanking movement. 3. Section A is finishing it's flanking movement and starts to approach the enemy. This also means that they are moving towards the location of where section B is providing fire. 4. The Raptor estimates the time they reach the fire zone of section B and when this time reaches the threshold value the Raptor notifies the PC. If he is watching at the raptor he will give the PC a salient visual cue to signal him that section B needs to be careful and soon needs to start diverting its fire. If he is not paying attention to the Raptor, the Raptor will give an audio alert. Raptor to Romeo: Section A will reach the firezone of section B in 15 seconds. Advised to let section B divert fire, over. 5. The PC contacts section B and orders them to divert fire. 6. Section A can safely approach and run over the enemy. Alternative steps 1.1 Section A is making a flanking movement and starts to approach the enemy. 1.2 Section B is providing support fire on the enemy in order to give A enough time to make the flanking movement. 1.3 Section A is finishing it's flanking movement and starts to approach the enemy. This also means that they are moving towards the location of where section B is providing fire. 1.4The Raptor estimates the time they reach the fire zone of section B and when this time reaches the threshold value the Raptor notifies the PC. If he is watching at the raptor he will give the PC a salient visual cue to signal him that section B needs to be careful and soon needs to start diverting its fire. If he is not paying attention to the Raptor, the Raptor will give an audio alert. Raptor to Romeo: Section A will reach the firezone of section B in 30 seconds. Advised to let section B divert fire, over. 1.5 The PC ignores the message and does not command section B to divert its fire. 1.6 Since the Raptor did not hear a command to section B he warns again. Raptor to Romeo: Acute risk on fratricide. Advised to let section B divert fire. If the PC is looking at the Raptor the audio message is still given due to its risk, and the visual cue is made even more salient. 1.7 The PC contacts section B just in time and they avert their fire just in time.

UC step	Requirements	Claims
4	RQ005: Predict a future risk of fratricide	CL004: The Raptor reassures that the PC will be aware of future risks of friendly fire
4	RQ011: Detect when the PC is looking at the Raptor	CL007: The Raptor reassures that the PC does not receive redundant notifications

UC005: No-go zone due to location of other platoon

Objective	OB01: Increase Situation Awareness of the Platoon Commander
Actors	SH01: Platoon Commander (PC)
Pre-condition	Section B made contact with the enemy and tagged their location. The Raptor predicted a future risk of fratricide based on the location of the enemy and the friendly platoon.
Post-condition	The PC makes an alternative plan.
Action sequence	1. The PC is planning an attack on the encountered enemy. The enemy is located on the forest edge of an open meadow. The PC sees a vegetated part of the location that enables one of his sections to make a flanking movement. 2. He consults his Raptor to check the map and to see if this indeed provides a viable attack option. After the enemy location is tagged, he sees that an area is marked on the maps and he understands that he cannot send his troops here. Apparently attacking from that position will lead to a risk of hitting friendly troops of the other platoon, of which the PC was not aware of their position. 3. The PC discards the plan and opts for a frontal approach using the bound overwatch technique. Alternative steps: 1.1 The PC is planning an attack on the encountered enemy. The enemy is located on the forest edge of an open meadow. The PC sees a vegetated part of the location that enables one of his sections to make a flanking movement. 1.2 Section B is providing support by fire and the PC is not giving them any waypoints. 1.3 When he plans the flanking movement of section A, he selects the section and draws the waypoints. When he selects section A he sees a no-go area behind the enemy. He understands that this no-go zone is created because of the fire support provided by section B and sees that is does not cause any trouble for his plan.

UC step	Requirements	Claims
2	RQ007: Create no-go zones	CL011: The Raptor reassures that the PC will not make a plan which will not plan an attack that will cause friendly fire with another platoon.

UC006: Using the IED prediction function in the planning phase

Objective	OB01: Increase Situation Awareness of the Platoon Commander
Actors	SH01: Platoon Commander (PC)
Pre-condition	The platoon made contact with an enemy force of which it is known that they sometimes place IEDs
Post-condition	The plan of the PC is avoiding COAs that include areas in which IEDs are placed
Action sequence	1. Based on the environment the PC sees a nice opportunity to flank the enemy. 2. He checks his Raptor to verify this possibility on the map. 3. The PC notices that the Raptor expects that the enemy placed IEDs on the route that the PC had in mind. 4. The PC comes up with a more difficult route to approach the enemy, but one on which there are no IEDs expected. He estimates that there is enough time available to approach the enemy using this route. 5. The PC communicates this plan plan to his sections Alternative: 1.1 Based on the environment the PC sees a nice opportunity to flank the enemy 1.2 He checks his Raptor to verify this possibility on the map 1.3 The PC notices that the Raptor expects that the enemy placed IEDs on the route that the PC had in mind. 1.4 The PC does not see another suitable route to approach the enemy and determines to communicate the original approach, even though the Raptor expects that the enemy has placed IEDs on that location. 1.5 When the PC draws in the waypoints, the Raptor lights up the IED that is on the path one more time, to make sure that the PC noticed the predicted IED. Alternative steps: 2.2 Based on the environment the PC sees a nice opportunity to flank the enemy. 2.2 He checks his Raptor to verify this possibility on the map. 2.3 The PC doesn’t see any IEDs on the map. He is not sure whether this means that there are no predicted IEDs or that the IED prediction function is turned off. 2.4 He checks in the menu wether the IED prediction functionality in turned on and sees that this is indeeed the case.

UC step	Requirements	Claims
3	RQ008: Predict likely locations for IEDs	CL006: The Raptor reasures that the PC validates self made predictions of IED locations

UC007: Alerting a section that approaches an expected IED

Objective	OB01: Increase Situation Awareness of the Platoon Commander
Actors	SH01: Platoon Commander (PC)
Pre-condition	The PC made a plan to attack the enemy and chose a flanking movement that crossed an area of which the Raptor expects IEDs to be present.
Post-condition	The section is warned and approaches the location carefully
Action sequence	1. Section A is making a flanking movement. The PC communicated that he expects the section to run into IEDs on their approach to the enemy. He gives them the approximate location. 2. Section A has made some progress in their approach and starts to get close to the expected IED. The PC is looking at the Raptor and therefore the Raptor only gives a visual signal that the section is within thresholding range. 3. The PC communicates this to Section A, which continues to move, but more carefully searching for signs of IEDs. Alternative steps: 1.1 Section A is making a flanking movement. The PC communicated that he expects the section to run into IEDs on their approach to the enemy. He gives them the approximate location. 1.2 Section A has made some progress in their approach and starts to get close to the expected IED. The eye tracker detects that the PC is not looking at the Raptor, even though the Raptor's screen is on. The Raptor sends an audio alert. Raptor to Romeo: section A is approaching expected IED. Advised to proceed with caution. 1.3 The PC looks at the Raptor to check the location of Section A and communicates to them that they are approaching an expected IED. 1.4 The section continues their approach but moves more carefully, looking for signs of IEDs. Alternative steps: 2.1 Section A is making a flanking movement. The PC communicated that he expects the section to run into IEDs on their approach to the enemy. He gives them the approximate location. 3.1 When section A is close to the predicted IED the PC communicates to them that they have to proceed with caution. 3.2 The Raptor understands this and does not contact the PC again to remind him to warn section A.

UC step	Requirements	Claims
2	RQ011: Detect when the PC is looking at the Raptor	CL008: The Raptor reassures that the PC stays in control of which information he receives
2	RQ009: Warn the PC when a section is approaching a suspected IED	CL010: The Raptor reassures that the PC will not miss a section approaching a predicted IED located.
2	RQ003: Send visual alerts	CL009: The Raptor reassures that the PC will always able to interact with the Raptor.

RQ001: Detect a section diverging from its commanded path

Specification	The Raptor should be able to detect when a section diverges from its communicated path.
Use cases	UC001: A flanking section starts to diverge from the communicated path

RQ002: Detect a section behind enemy lines

Specification	The Raptor should be able to detect when a section is moving too far and ends up behind enemy lines.
Use cases	UC002: A flanking section ends up behind the enemy line

RQ003: Send visual alerts

Specification	When the PC is looking at the Raptor while one of the functionalities detects an alarming situation the Raptor should present the PC with a visual cue.
Use cases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC003: Acute risk of fratricide UC007: Alerting a section that approaches an expected IED

RQ004: Detect an acute risk on fratricide

Specification	The Raptor should detect when there exists an acute risk on fratricide
Use cases	UC003: Acute risk of fratricide

RQ005: Predict a future risk of fratricide

Specification	The Raptor should be able to predict a future risk of fratricide.
Use cases	UC004: Future risk of fratricide

RQ006: Include buildings in determining risk of fratricide

Specification	The Raptor should take building into account when determining and predicting the risk of fratricide since buildings will stop bullets.
Use cases	UC003: Acute risk of fratricide

RQ007: Create no-go zones

Specification	The Raptor should be able to draw no-go zones on the map based on the plans of other sections and the locations of other platoons. These no-go zones represent areas that will likely lead to a risk of fratricide.
Use cases	UC005: No-go zone due to location of other platoon

RQ008: Predict likely locations for IEDs

Specification	The Raptor should be able to predict the most likely locations for the enemy to place IEDs. This prediction is made based on the locations of buildings.
Use cases	UC006: Using the IED prediction function in the planning phase

RQ009: Warn the PC when a section is approaching a suspected IED

Specification	The Raptor should be able to detect when a section is approaching a location where an IED is expected. This way it can be made sure that the section is approaching this location with caution.
Use cases	UC007: Alerting a section that approaches an expected IED

RQ010: Limit the number of alarms

Specification	The Raptor should limit the number of notifications that it gives to the PC. Only useful notifications should be communicated and redundant notifications should be prevented.
Use cases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC003: Acute risk of fratricide UC004: Future risk of fratricide UC007: Alerting a section that approaches an expected IED

RQ011: Detect when the PC is looking at the Raptor

Specification	Detect when the PC is looking at the Raptor
Use cases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC004: Future risk of fratricide UC007: Alerting a section that approaches an expected IED

RQ012: Turn off individual functionalities

Specification	The PC should be able to turn off individual functionalities. This keeps him in control of the notification that he wishes to receive.
Use cases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC006: Using the IED prediction function in the planning phase

RQ013: Show the alerts of a single section

Specification	It should be possible for the PC to select a single section and see only the alerts that relate to this section.
Use cases	UC001: A flanking section starts to diverge from the communicated path

RQ014: Understand voice commands

Specification	The Raptor should be able to understand voice commands. This enables the PC to interact with the Raptor when the Raptor is tucked away.
Use cases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC003: Acute risk of fratricide

RQ015: Send audio alerts

Specification	The Raptor should be able to send audio alerts in case the PC has not been looking at the Raptor for a while. This way the PC will still be able to get updated about important situations.
Use cases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC003: Acute risk of fratricide UC004: Future risk of fratricide UC007: Alerting a section that approaches an expected IED

CL001: The Raptor reassures that the PC will be aware of sections that diverge from their planned route

Ups ide	The Raptor will send the PC a notification when a section is diverting from its planned route. This will make the PC more confident about not missing these cues due to a high cognitive workload.
Dow nside	The alert that this functionality pushes can be conceived as annoying. It may also occur that the PC does not fully trust the Raptor, making the notification lose its benefit and only causing extra distraction.
Use Cas es	UC001: A flanking section starts to diverge from the communicated path

CL002: The Raptor reassures that the PC will be aware of sections that end up behind enemy lines

Up side	The Raptor will send the PC a notification when a section is moving behind enemy lines. This will make the PC more confident about not missing these cues due to a high cognitive workload.
Do wn side	The notification that this functionality pushes can be conceived as annoying. It may also occur that the PC does not fully trust the Raptor, making the notification lose its benefit and only causing extra distraction and workload.
Us e cas es	UC002: A flanking section ends up behind the enemy line

CL003: The Raptor reassures that the PC will be aware of occurrences of acute risks of friendly fire

Up side	The Raptor will send the PC a notification when an acute risk of friendly fire occurs. This will make the PC more confident about not missing these cues due to a high cognitive workload.
Do wn side	The notification that this functionality pushes can be conceived as annoying. It may also occur that the PC does not fully trust the Raptor, making the notification lose its benefit and only causing extra distraction and workload.
Us e cas es	UC001: A flanking section starts to diverge from the communicated path UC003: Acute risk of fratricide

CL004: The Raptor reassures that the PC will be aware of future risks of friendly fire

Upside	The Raptor will send the PC a notification when it predicts a future risk of friendly fire. This will make the PC more confident about not missing these cues due to a high cognitive workload.
Downside	The notification that this functionality pushes can be conceived as annoying. It may also occur that the PC does not fully trust the Raptor, making the notification lose its benefit and only causing extra distraction.
Use cases	UC002: A flanking section ends up behind the enemy line UC004: Future risk of fratricide

CL005: The Raptor reassures that te PC creates a plan that anticipates on the presence of IEDs

Upside	The PC might feel more confident about the created plan when the PC can take the predicted locations of IEDs into account.
Downside	The PC might start to trust the IED prediction too much. Because of its nature, the prediction will by definition not always predict the locations of the IEDs correctly. If the PC relies too much on the predictions of the Raptor, this feeling of confidence might be misplaced.
Use cases	UC006: Using the IED prediction function in the planning phase

CL006: The Raptor reasures that the PC validates self made predictions of IED locations

Upside	Based on experience and gut feeling a PC will expect IEDs in certain parts of the environment. The IED predictions of the Raptor allow the PC to validate these predictions.
Downside	The PC might be trusting his own predictions or the prediction of the Raptor too much. This can give him a misplaced feeling of SA and make the PC too reluctant in looking for IEDs in locations where no IED was predicted.
Use cases	UC006: Using the IED prediction function in the planning phase

CL007: The Raptor reassures that the PC does not receive redundant notifications

Upside	This functionality will make sure that the PC does not receive audio notifications while he is looking at the Raptor. This will reduce his mental workload and prevents alert fatigue. At the same time features in the implementation, like taking buildings into account when calculating the risk of friendly fire, should prevent false alarms as much as possible
Downside	It might be unclear for the PC when he receives an auditory message and when not. Furthermore, the sensitivity needed to prevent redundant notifications and false alarms can overshoot and cause false negatives: mission-critical cues that are not communicated to the PC.
Usecases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC004: Future risk of fratricide

CL008: The Raptor reassures that the PC stays in control of which information he receives

Upside	When the PC is able to turn certain notifications off, he is able to focus on a certain problem better. Furthermore, it will keep the PC in control of what information he receives and what is not important at this moment. This will also enable the PC to obtain the information about a single section when he is interested in this section in particular.
Downside	When a PC turned off a single functionality he might not notify the occurrence of an important situation. This can also happen when the PC turned a functionality off but forgot to turn it back on again. Focussing on a single section might cause the PC to miss alarming situations occurring for another section.
Usecases	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC007: Alerting a section that approaches an expected IED

CL009: The Raptor reassures that the PC will always able to interact with the Raptor.

Up side	This will make sure that the PC becomes able to interact with the Raptor, even when the Raptor is tucked away. This happens when the PC is moving for example.
Do wn side	The solution for this will likely be that the Raptor will include more modalities to interact with the PC. This could cause an overload if these modalities are already used by the PC, for example, audio for radio communication.
Use cas es	UC001: A flanking section starts to diverge from the communicated path UC002: A flanking section ends up behind the enemy line UC003: Acute risk of fratricide UC004: Future risk of fratricide UC006: Using the IED prediction function in the planning phase UC007: Alerting a section that approaches an expected IED

CL010: The Raptor reassures that the PC will not miss a section approaching a predicted IED located.

Ups ide	The Raptor will send the PC a notification when a section is approaching an IED. This will make the PC more confident about noticing these situations.
Do wns ide	The notification that this functionality pushes can be conceived as annoying. It may also occur that the PC does not fully trust the Raptor, making the notification lose its benefit and only causing extra distraction.
Use cas es	UC001: A flanking section starts to diverge from the communicated path UC007: Alerting a section that approaches an expected IED

CL011: The Raptor reassures that the PC will not make a plan of attack that will cause friendly fire with another platoon.

Upside	The Raptor knows the locations of friendly platoons as well. When an enemy is drawn on the map the Raptor will indicate an area from which the PC should not flank the enemy. Flanking from this direction would result in a risk of inducing friendly fire with a friendly platoon.
Downside	Coordinating this is the task of the Squadron Commander. This extra information might clutter the display of the Raptor
Use cases	UC005: No-go zone due to location of other platoon

Appendix F: Heuristic evaluation results

Identified usability issues

Simple and Natural Dialogue

1. Red is used a lot in the warning symbols. When these warnings happen close to an enemy unit (which is also red), the symbols can easily overlap and the red color can lose its salient function.
2. The unfolding menu structure takes up a big part of the screen.
3. The symbol for diverging from the plan is unclear.
4. The symbol to open the menu of added functionalities relates to how this function works (automatically), instead of what the functionality actually does: giving alerts.
5. The action of holding a button to open up another functionality is not yet used anywhere else in the Raptor.
6. The audio notifications use a lot of formal language.
7. The use of the text *"B.O.B."* in the fratricide risk icon is unclear.

Speak the user's language

-

Minimize the user's memory load

8. Due to the shape of the triangle it is ambiguous on which way a section is looking that is not moving.
9. The function that monitors the execution of the plan shows the time that it takes a diverged section to return to the plan. The important information here is the effect on its ETA on the location from where it can attack the enemy, but this is not shown.
10. The WDA in the visual alert for a future risk of fratricide is not salient enough.
11. The progress bar that indicated how long a monitoring function is still turned off does not add extra information
12. When a section has a future risk on fratricide with two other sections, only the most urgent is made salient.
13. The display will be very cluttered when multiple alarms go off at the same time.
14. Audio notifications are presented via the voice channel, which is normally occupied a lot.

Be consistent

15. The red arrow in the warning that a section ended up behind enemy lines is not consistent. Red normally means danger or do not do this, however, here it means: go this way as quickly as possible.
16. The triangle with the exclamation mark is also used in the interface for tagging an enemy and therefore has two different meanings.
17. The location of a section is indicated differently from the location of an enemy.
18. The grid lines of the map look like the line that is used to indicate the border of the enemy line.
19. Important areas, like the no-go zone and the WDA's during a (future) risk of fratricide, are displayed in different ways.
20. The concerned section is not highlighted in the warning of a section moving behind enemy lines
21. A red dashed line is used to indicate the enemy front line and to indicate the relevant edge of the WDA when there is a risk of fratricide.

22. The no-go zone is not always visible when the enemy is drawn.
23. No suggestions to solve a problem are given, except for when a section ends up behind the enemy line.
24. The red color of the warning symbols can create confusion as red is normally a color that is used to indicate enemy groups.
25. The bot menu does not open up on the side like the other functionalities.
26. In some situations an increase in size is related to importance while in other situations a change in opacity implicates importance.
27. A section gets outlined by a red line when it is selected and when an alert is given for this section. However the red outline does not mean the same thing.

Provide feedback

28. The Raptor does not reply to a voice command of the commander. Therefore the commander does not know if the Raptor understood him and if the settings are changed or not.

Provide clearly marked exits

-

Provide shortcuts

29. When the Raptor presents the salient view for ten seconds, there is no way that the commander can return to the overview quicker.
30. There is no easy way to remove all waypoints for a section.
31. It is not possible to move around objects like waypoints or an enemy location.

Provide good error messages

32. The Raptor does not give any error messages. It can happen that the Raptor does not understand the command, but it will not communicate this to the PC.

Prevent errors

33. When the waypoint button is pressed and a section is selected, the other sections do not grey out. This can lead to a PC entering waypoints for the wrong section.
34. When a waypoint is placed, the Raptor does not show the PC how far a section can diverge from this waypoint without receiving an alert.
35. The Raptor does not warn the PC when he tries to place a waypoint behind the enemy line.

ISSUE	EV1	EV2	EV3	EV4
1	x			
2		x		
3		x		
4		x		
5			x	
6			x	
7				x
8	x	x		
9	x			
10	x	x		
11		x		
12		x		
13			x	
14				x
15	x			
16		x	x	
17		x		
18		x		
19			x	x
20			x	
21			x	
22			x	
23			x	
24	x			
25				x
26				x
27				x
28	x	x		x
29	x			x
30		x		
31		x		x
32	x			
33	x			
34		x		
35		x	x	
Issues identified:	*11/38*	*15/38*	*10/38*	*9/38*

Appendix G: Questionnaire used during the user test

Mission Awareness Rating Scale (MARS)

The first four questions deal with the ability *of the Raptor to support in the detection and understanding of important cues present during the mission.*

1. Please rate how well (expected) the Raptor helped to ***identify*** mission critical cues in the mission
   - ☐ very well - able to identify all cues
   - ☐ fairly well – could identify most cues
   - ☐ somewhat poorly - many cues hard to identify
   - ☐ very poorly - had substantial problems identifying most cues

2. Please rate how well the Raptor helped to ***understand*** mission critical cues in the mission
   - ☐ very well - fully understood the situation as it unfold
   - ☐ fairly well - understood most aspects of the situation
   - ☐ somewhat poorly - had difficulty understanding much of the situation
   - ☐ very poorly - the situation did not make sense to me

3. Please rate how well the Raptor helped to ***predict*** mission critical cues in the mission
   - ☐ very well - could predict with accuracy what was about to occur
   - ☐ fairly well - could make accurate predictions most of the time
   - ☐ somewhat poorly - misunderstood the situation much of the time
   - ☐ very poorly – unable to predict what was about to occur

4. Please rate how well the Raptor helped in deciding ***how to best achieve*** the goals of this mission
   - ☐ very well - knew how to achieve goals at all times
   - ☐ fairly well - knew most of the time how to achieve mission goals
   - ☐ somewhat little - was not aware of how to achieve some goals
   - ☐ very little - generally unaware of how to achieve goals

The last four questions deal with the mental effort *required to detect and understand important cues present during the mission by using the Raptor.*

5. How much helped the Raptor in reducing the mental effort needed for **identifying** or detecting mission-critical cues in the mission?
   - ☐ very much - using the Raptor made me identify relevant cues with little effort
   - ☐ fairly much – using the Raptor made me identify relevant cues, but with some effort required
   - ☐ somewhat little - using the Raptor took some mental effort to identify most cues
   - ☐ very little - using the Raptor took me substantial effort to identify relevant cues

6. How much helped the Raptor in reducing the mental effort needed for **understanding** what was going on during the mission?
   - ☐ very much - using the Raptor made me understood what was going on with little effort
   - ☐ fairly much - using the Raptor made me understood events with only moderate effort
   - ☐ somewhat little - even with the Raptor some aspects of the situation were hard to comprehend
   - ☐ very little - even with the Raptor most or all aspects of the situation were hard to understand

7. How much helped the Raptor in reducing the mental effort needed for **predicting** what was about to happen during the mission?
   - ☐ very much - using the Raptor made me predict what was about to happen with little or no effort
   - ☐ fairly much - using the Raptor made me predict what was about to happen with moderate effort
   - ☐ somewhat little – using the Raptor, substantial effort was required for many prediction
   - ☐ very little – using the Raptor, substantial effort was required on most or all prediction.

8. How much helped the Raptor in reducing the mental effort needed in deciding ***how to best achieve*** mission goals during this mission?
   - ☐ very much - using the Raptor, no or little effort was needed for all decisions
   - ☐ fairly much - using the Raptor, moderate effort was required for most decisions
   - ☐ somewhat little – using the Raptor, substantial effort was needed for some decision
   - ☐ very little – using the Raptor, most or all decisions required substantial effort.

Usability evaluation of current implementation

9. The *meaning* of the alerts from the … functionality was very clear to me

	1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree
IED	1	2	3	4	5
Route monitoring	1	2	3	4	5
Fratricide detection	1	2	3	4	5
Fratricide prediction	1	2	3	4	5

Comments:

10 I found the *amount of* alerts that given by the … functionality too high

	1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree
IED	1	2	3	4	5
Route monitoring	1	2	3	4	5
Fratricide detection	1	2	3	4	5
Fratricide prediction	1	2	3	4	5

Comments:

10 I found *the way in which* alerts were given by the … functionality pleasant

	1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree
IED	1	2	3	4	5
Route monitoring	1	2	3	4	5
Fratricide detection	1	2	3	4	5
Fratricide prediction	1	2	3	4	5

Comments:

Applicability evaluation in a real scenario

10. I believe that the function that predicts the locations and likelihood of IEDs will benefit Platoon Commanders on a real battlefield.

1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree

I would use this technology slightly different so it would be more suitable on a real battlefield, namely to / in the following way:

11. I believe that the function that monitors whether sections are following the communicated route will benefit a Platoon Commander on a real battlefield.

1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree

I would use this technology slightly different so it would be more suitable on a real battlefield, namely to / in the following way:

12. I believe that the function that monitors the acute risks of fratricide will benefit a Platoon Commander on a real battlefield.

1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree

I would use this technology slightly different so it would be more suitable on a real battlefield, namely to / in the following way:

13. I believe that the function that predicts the risks of fratricide in the near future will benefit a Platoon Commander on a real battlefield.

1 = Strongly disagree	2 = Disagree	3 = Neutral	4 = agree	5 = strongly agree

I would use this technology slightly different so it would be more suitable on a real battlefield, namely to / in the following way: