



EXAMENSARBETE INOM ELEKTROTEKNIK,
GRUNDNIVÅ, 15 HP
STOCKHOLM, SVERIGE 2017

Evaluation of HCI models to control a network system through a graphical user interface

Utvärdering av MMI-modeller för styrning av
nätverkssystem genom grafiskt
användargränssnitt

JONAS EKMAN

Evaluation of HCI models to control a network system through a graphical user interface

Utvärdering av MMI-modeller för styrning av nätverkssystem genom grafiskt användargränssnitt

Jonas Ekman

Degree project in Electrical Engineering
Ground level: 15 hp
Supervisor at KTH: Anders Cajander
Examiner: Thomas Lind
School of Technology and Health
TRITA-STH 2017:41

Royal Institute of Technology
School of Technology and Health
Hälsövägen 11C, 141 57 Huddinge
www.kth.se/sth

Abstract:

SAAB has a project under development for a network system with connected nodes, where the nodes are both information consumer and producer of different communication types. A node is an equipment or an object that are used by the army e.g. it can be a soldier, military hospital or an UAV. The nodes function as a part of a mission e.g. a mission can be *Defend Gotland*. The aim of this project is that the user will rank different missions from the one with the highest priority to the lowest. This will affect the network in a way that the communication between the nodes at the mission with the highest rank will be prioritised over the communication between the underlying missions. A user can via the GUI rank the missions, and then set the associated settings for them. Via the GUI the user should be able to work at three different levels. The first is to plan upcoming missions. The second one is in real time see if the system delivers the desired conditions. The last one is to simulate if the system can deliver the desired conditions.

This thesis investigated various HCI models that could be used to create a GUI, to reduce the risk of a user configuring the system incorrectly. The study showed that there are no HCI models that take account for misconfigurations, and therefore a new model was created. The new model was used and evaluated by creating a prototype of a GUI for SAAB's project and was tested on a potential user. The test showed that the new model reduced the risk of misconfigurations.

Keywords: GUI, Graphical user Interface, HCI, Human-computer Interaction, HCI models, Military Symbols and Military Systems.

Sammanfattning:

SAAB har ett projekt för utveckling av ett nätverkssystem med anslutande noder, med noder som kan vara både informationsproducent och konsument för olika kommunikationstyper. En nod är en sak eller ett objekt inom försvaret t.ex. kan det vara en soldat, militärt sjukhus eller en obemannad farkost. Varje nod tillhör ett uppdrag, tex att försvara Gotland. Målet med projektet är att man ska kunna gradera de olika uppdragen och därmed gradera vilken prioritet dessa noder har i nätet. Noder som tillhör ett uppdrag med hög gradering kommer prioriteras över de underliggande uppdragen i nätverket. En användare kan via ett grafiskt användargränssnitt gradera uppdragen och konfigurera tillhörande inställningar. Via det grafiska användargränssnittet kan en användare även planera, gradera och konfigurera inställningar för kommande uppdrag samt simulera om det går att genomföra. Användaren ska även i realtid kunna se om de önskade inställningarna inte kan leva upp till de önskade kraven, och därmed kunna åtgärda detta.

Detta arbete undersökte olika MMI-modeller som kan användas för att skapa ett grafiskt användargränssnitt som minimerar risken att användaren konfigurerar systemet på ett felaktigt sätt. Studien visade sig att det inte finns några MMI modeller som tar hänsyn till felkonfigureringar, och en ny modell skapades. Den nya modellen användes och utvärderas genom att skapa en prototyp av ett grafiskt användargränssnitt för SAAB:s projekt, som testades på en potentiell användare. Testet visade att den nya modellen minskar risken för felkonfigureringar.

Nyckelord: Grafisk användargränssnitt, Människa-maskin interaktion, MMI, MMI-modeller, Symboler för militärer och militära system.

Acknowledgements:

This thesis is the final exam in bachelor program in electrical engineering at the Royal Institute of Technology.

I would like to thank SAAB and the department of Combat Systems and C4I Solutions for giving me the opportunity to investigate this thesis. I would also like to thank my supervisor Kjell Svensson at SAAB for his support and contribution of knowledge to this project.

I would also like to thank my supervisor Anders Cajander at KTH for giving me advice and guidance on the thesis as well as giving me important feedback on the report.

Jonas Ekman
Stockholm, June 2017

Abbreviations

AI – Artificial intelligence

BFT – Blue Force Tracking

Control – System Management traffic, e.g. authorities.

COP – Common Operational Picture

GOMS – Goals, Operators, Methods and Selection Rules

GUI – Graphic user interface

HCI – Human-computer Interaction

HRI – Human-robot interaction

IoT – Internet of Things

ISR – Intelligence, surveillance and reconnaissance

LTM – Long term memory

Msg – Command and control messages

NGOMSL – Natural GOMS language

Node – Things/Equipment connected to the network

OAI – Object action interface

P_2_P – Point to point communications, is a task the user can do in the graphical user interface.

UAV – Unmanned aerial vehicle

Video – Streamed Video

Voice – Streamed Voice

Table of contents

1.	Introduction	1
1.1.	Background.....	1
1.2.	Problem definition	1
1.3.	Goal	1
1.4.	Delimitations	2
1.5.	Related works	2
2.	Theory and background	3
2.1.	Background information about the project	3
2.1.1.	Nodes	4
2.1.2.	Requirements for the GUI	5
2.2.	Human-computer interaction (HCI)	5
2.2.1.	The different modes in HCI	5
2.2.1.1.	Mode 1 Data interaction	5
2.2.1.2.	Mode 2 Image interaction	6
2.2.1.3.	Mode 3 Voice interaction.....	6
2.2.1.4.	Mode 4 Intelligent interaction	6
2.3.	The design process for a human computer interface	6
2.3.1.	The Seeheim model for structural design	7
2.3.2.	Behavioural model	8
2.3.2.1.	Goals Operators Method and Selection (GOMS) model.....	8
2.3.2.2.	Enhanced OAI Model.....	9
2.3.2.3.	NGOMSL	10
2.3.2.4.	Fitts's Law.....	12
2.3.3.	Important design aspects.....	13
2.3.3.1.	Analysis of the users	13
2.3.3.2.	Ergonomics.....	14
2.3.3.3.	Reducing the risk of user mistakes	14
2.3.3.4.	Identifiable and operational design	14
2.3.3.5.	Communicating between user and computer	14
2.3.3.6.	Shortcuts.....	14
2.3.3.7.	Feedback.....	14
2.3.3.8.	Short term memory.....	15
2.3.3.9.	Help function	15
2.4.	Virtual design	15
2.4.1.	Light	16
2.4.2.	Military symbols.....	16
2.4.2.1.	Colour	17
2.4.2.2.	Identities and dimensions	17
2.4.2.3.	Icon	18
2.4.2.4.	Modifiers	18
3.	Methods and result	19
3.1.	New HCI model	19
3.1.1.	Task and goal.....	19

3.1.2.	Structure	19
3.1.3.	Design.....	20
3.2.	Solution methodology	21
3.3.	Prototype of the GUI with the new HCI model.....	21
3.3.1.	Method to calculate the Fitts's law	22
3.3.2.	Planning mode.....	22
3.3.3.	Set priority for the missions.....	22
3.3.3.1.	Task and goal	22
3.3.3.2.	Structure	23
3.3.3.3.	Design	25
3.3.4.	Edit properties for the desired mission.....	27
3.3.4.1.	Task and goal	27
3.3.4.2.	Structure	28
3.3.4.3.	Design	30
3.3.5.	Point to Point	34
3.3.5.1.	Task and goal	34
3.3.5.2.	Structure	36
3.3.5.3.	Design	38
3.3.6.	Live mode.....	41
3.3.7.	Select Mission	41
3.3.7.1.	Task and goal	41
3.3.7.2.	Structure	43
3.3.7.3.	Design	44
3.3.8.	Point to Point.....	46
3.3.9.	Simulation mode	47
3.4.	Test on potential user	47
4.	Analysis and discussion.....	49
4.1.	Choice of HCI model	49
4.2.	New HCI model	49
4.2.1.	Task and goal	50
4.2.2.	Structure	50
4.2.3.	Design.....	50
4.3.	Solution methodology	51
4.4.	Test of GUI.....	51
4.5.	Sustainable development	51
5.	Conclusion.....	53
6.	Continuous work.....	55
7.	Appendix	57
7.1.	Appendix A.....	57
7.2.	Appendix B.....	58
7.3.	Appendix C.....	59
8.	Reference	61

1. Introduction

This thesis has been carried out on behalf of the C4I Solutions and Combat Systems at SAAB.

1.1. Background

SAAB is one of the leading providers of military equipment and develops everything from air fighters to submarines. Their fighter jet JAS 39-gripen is one of their, for the public, most known products. At SAAB, they have a department called C4I solutions and Combat system, which is specialised in developing surveillance systems. One of the projects that are under development is a network system, where the nodes are different military equipment e.g. UAV, radar and military hospital. The nodes are both producer and consumer of different communication types e.g. video and tracking. The idea is that under some special situation some types of communication types will be more prioritised and have greater quality compared to other communication types. The system will be operated through a GUI in which an operator will make these settings.

1.2. Problem definition

The problem is that there is a risk of misconfigurations within a GUI which is not designed appropriately. For systems which process important information, this can have significant consequences on the system, and reduce system efficiency. In projects such as those performed by SAAB, a misconfiguration can have significant impact on how the military missions will be performed. The consequences could be that the mission fail to perform their task. This could have a significant impact the country's safety, and perhaps also lead to wounded soldiers or even casualties.

To find a solution for this problem, this paper investigates different HCI models for analysing the connection between the system and GUI and also between the GUI and the user. Creating a GUI which represents the system correctly and reduces the risk of a user misconfiguring the system.

1.3. Goal

The goal of this thesis is to investigate different theoretical HCI model or create a new model, that can be used in developing the GUI. The HCI model is created in a way that it will reduce the risk of the user to configure the system incorrectly. From the chosen HCI model, a GUI will be created, and then tested on a potential user.

1.4. Delimitations

This thesis has been limited to investigating of human-computer interaction for a graphical user interface. There the focus is for an operator with knowledge about the system, but lacks knowledge about the implementation and technical aspects of the system.

- The design of the GUI will be created by predefined functions in JavaFX.
- No interviews with potential users will be held, due to that the analysis of the HCI models is believed to give a sufficient good image on how the GUI should be designed to increase the usability.
- HCI models will be analysed from the perspective of a user who will use the GUI from a computer with mouse and keyboard, and not by other methods e.g. touchscreen.
- The project had to be carried out over 10 weeks between the end of March 2017 to the end of May 2017.

1.5. Related works

There are no other similar products available on the market today that has the same functionality as the one which is under development at SAAB.

For HCI has more scientific researchers been done, in how to make the complex system useful, and interactive to the user. One study was to investigate the HCI model GOSM for human-robot interaction. The study resulted in that the NGOSML model was the most suitable for HRI [1]. Another study was to investigate different HCI models for Internet of things and more specific smart homes. To find a model that could enhance the usability and the interaction to the user. They find that the most suitable model was R-GOMS model, because it had predefined steps for analysing user information and to understand the user needs [2].

2. Theory and background

This section presents the background theories for the project at SAAB and HCI.

2.1. Background information about the project

The equipment that is used by the army is represented as nodes in this project, e.g. a node can be military hospital or an UAV. At this state, there are only 8 different nodes represented, see table 1, but the system should be able to operate with more nodes. The idea is that under some situation, the army needs to be able to choose what kind of communication type or nodes that should be prioritised, e.g. video, BFT or military hospital. A real situation can be to evacuate wounded soldiers, a situation where video and voice communication between the medical and the soldier should have higher quality and priority in the network compared to the other nodes and communication types. Due to limitation of bandwidth of the backbone network, all the different communication types which are in all the individual nodes cannot produce their service at the highest quality.

The nodes belong to a mission and are predefined from the order of combat. An operation is a mission to do something e.g. capture the hill or medical evacuation of wounded soldiers, and an operation belongs to an organisation e.g. defend Gotland.

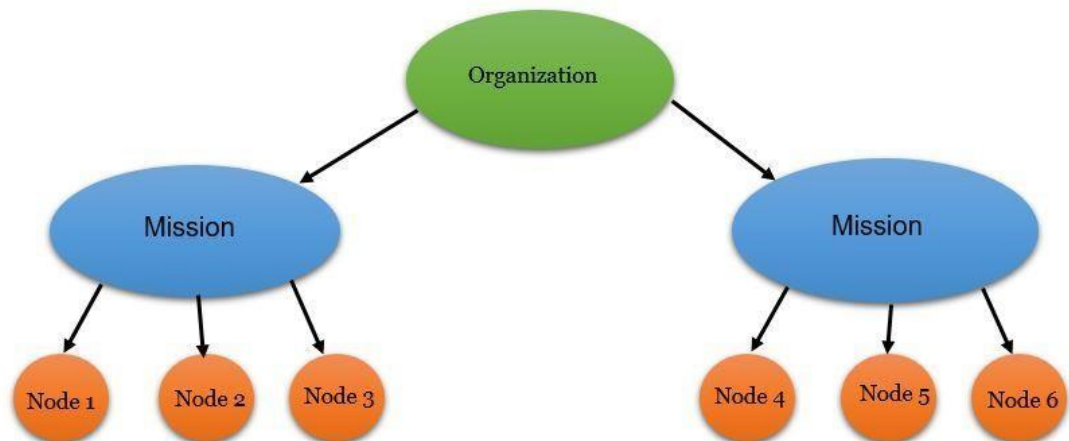


Figure 1 Tree of an organisation structure

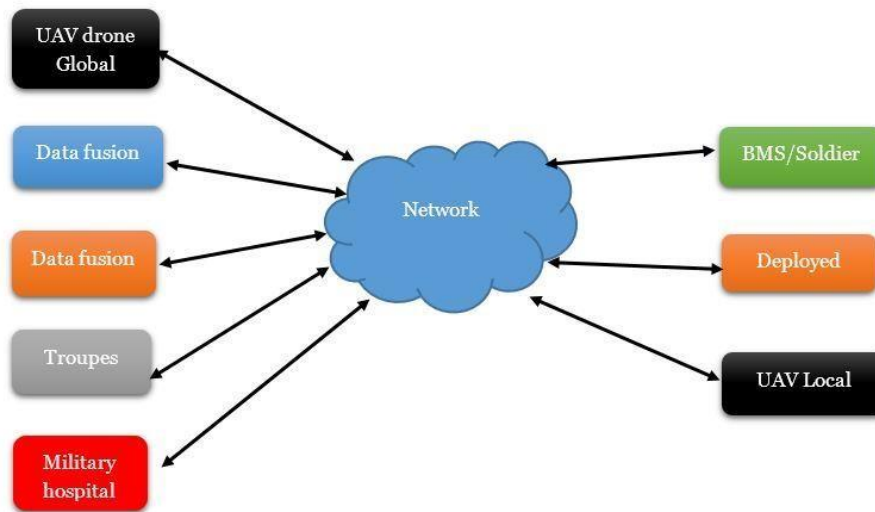


Figure 2 Potential network configuration

2.1.1. Nodes

In this stage of the project, there are only 8 different nodes, and all of them are consumer and producer of different types of communications, e.g. voice, video or BFT. The final version of the system will contain more nodes. At this state, the 8 nodes are as follow, see table 1.

Table 1 Table for the consumer (C) and producer (P) of the different communication types in the nodes. The green colour means the node is producer or consumer, and the red means it is not.

Node/Communication type	BFT	COP	Voice	ISR	Video	Msg	Control
UAV Global	P	C	P	C	P	C	P
Data Fusion M	P	C	P	C	P	C	P
Data Fusion S	P	C	P	C	P	C	P
Troupes	P	C	P	C	P	C	P
Military Hospital	P	C	P	C	P	C	P
Soldier	P	C	P	C	P	C	P
Deployed	P	C	P	C	P	C	P
UAV Local	P	C	P	C	P	C	P

2.1.2. Requirements for the GUI

The requirements of the GUI are that the operator will be able to work at three different levels:

- 1) **Plan:** the operator will be able to plan upcoming mission,
- 2) **Live:** At this level, the user will be able to get an overview of how the system operates in real time, and with nodes that are not operating at to the desired conditions.
- 3) **Simulate:** at this level, the operator will test the settings for the nodes and communication types and simulate if the network can deliver it, or if it needs more network equipment or reconfiguration.

The following requirement is for all the different levels:

- Set priority and quality between two nodes in a mission.
- Set priority and quality for the different communication types in a mission.
- Set the priority the for the different nodes in a mission.
- Set the priority for the different missions.

2.2. Human-computer interaction (HCI)

An important part in creating a graphic user interface is the study about the relationship between the machine and human (HCI, human-computer interaction). The point of this is to find communication methods that makes the input and output from the computer easy for the operator to use. It also reduces the connectivity load of the user greatly and increases the usability of the system [3].

2.2.1. The different modes in HCI

The implementation HCI in GUI makes the information flow between the system and the user harmonically. The user sends instructions to the system through a GUI, and then the system respond after processing. The information between the user and the system can be various forms e.g. data interaction, image interaction, voice interaction and intelligent interaction [3].

2.2.1.1. Mode 1 Data interaction

Data interaction has an important role in HCI, it is about putting data into the computer and exchange it for information. Most of the time, the process between the human and computer: is that the computer asks the user for an input of data; then in respond the system generates feedback to the user which may be presented on the screen. There are many methods for entry of data to a system. There are also different ways of interactive data, there the data can be all kinds of information symbols e.g. figure, graphics and colours [1].

2.2.1.2. **Mode 2 Image interaction**

People can interact with others through messages in three different ways: language, words and images. There 70% of the obtained information from visual systems is from images. Therefore, images have a significant role in interaction with people [1].

2.2.1.3. **Mode 3 Voice interaction**

Voice interaction is the interaction between the computer and people or other information facilities. It is usually a two-way communication [3].

The first one is with a system what uses voice recognition and understanding technology, which depends on the interactive audio system. The second one is the use of audio or voice to communicate to user e.g. it can be success or failure [3]. From different studies, it has been proven that the auditory signal detection is faster than the detection from virtual signals. Because of that, auditory signals are the most important information channel between the computer and the user [1].

2.2.1.4. **Mode 4 Intelligent interaction**

The implementation of AI in GUI will be the next generation of technology within HCI. This could lead to a scenario where the machine can communicate to the user by voice and data interaction and will learn from the behaviour of the user, and adapt to the needs of the user [3].

2.3. **The design process for a human computer interface**

The design process of HCI for GUI can be developed from some different perspectives. If the study in HCI is from the perspective of the structure of the system, it is called structural model e.g. Seeheim model. The structural is most of the time divided into three categories: specification, dialogue control and application interface [4]. If the study is from analysing the potential users' reaction and characteristics, then is are called behavioural model [5] e.g. GOMS and Enhanced OAI –model.

Different models are described in the following section. These models are chosen because they are well documented and has been used in other scientific research, and may have potential for solving the problem.

2.3.1. The Seeheim model for structural design

The Seeheim model is a structure based model and the perspective of the development is the structure of the system and not the user. Developing process of the model is often divided into 3 steps [4].

- **Presentation:** This state is to define the internal mapping of basic symbols. The input from the user is translated to numbers and basic symbols to create a dialogue between the user and the system [4].
- **Dialogue Control:** This state is to define the structure of the dialogue between the user and the application program. It also has the responsibility for routing the basic symbol to the appropriate part of the system/application [4].
- **Application interface:** This state is the representation of the application. It defends the semantics between the application and data object that are relevant to the user interface and the associated action it is related to. It has the responsibility to communicate between the application by its requirement to the user interface [4].

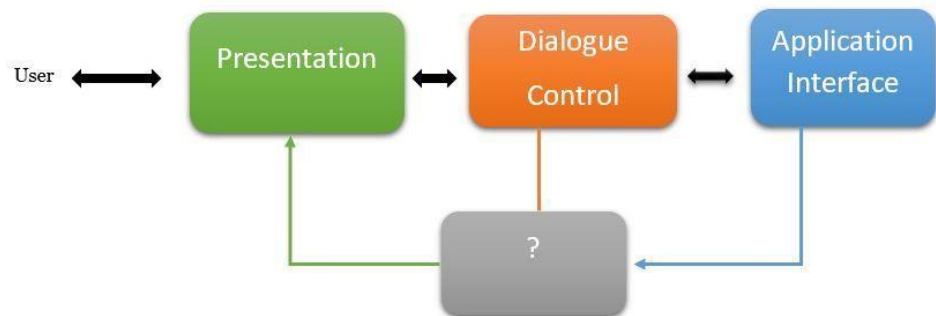


Figure 3 Seehiem model

The box marked “?” in figure 3, allows rapid semantic feedback. An example of that can be freehand drawing and highlighting the trash bin on Apple Macintosh when a file is over the symbol [6].

2.3.2. Behavioural model

The behavioural model is by studying the potential users' reactions and characteristics. There are a lot of different models to use for analysing the user, this paper is only describing a few of them.

2.3.2.1. Goals Operators Method and Selection (GOMS) model

The GOMS model is a method used to analyse the user's knowledge of how to use a special task, in terms of goals, operations, methods and selection [7]. GOMS model analysis can also be done when a company want to buy new software program or creating a new software. There are many examples of companies that have saved millions of dollars, e.g. NYNEX who used GOMS before buying new software, and the conclusion was that the new software was inefficient compared to the program they used [6].

Goals: Goals refers to analysing the goals of the system, in terms of what does he or she want to accomplish by using the software? Does the goal need to be accomplished in the next day, the next few minutes or the next few seconds? [7].

Operations: Operations often refers to the cognitive process and the physical actions the user needs to perform to accomplish the goal. If the goal is to search for an object in a search engine, the user must choose a specific search engine and then enter a keyword to the search engine [8].

Methods: This section is to analyse well-learned sequences. The well-learned sequence is when the user is using sub-goals and operations to accomplish a goal. The classic example of this is when you are deleting a text. You can place the cursor at the beginning of the text, hold the mouse button down and drag it to end of the text, and then hit the delete key. It can also be accomplished by placing the cursor at the end of the text and then hit the delete key [7].

Selection: Selection role is when there is more than one way to accomplish the goal. For example, when searching within search engines and the keyword have been entered in the entry field. There are many search engines that allow the user to press either the enter key or a go button. A selection rule would determine when that method that should be used [8].

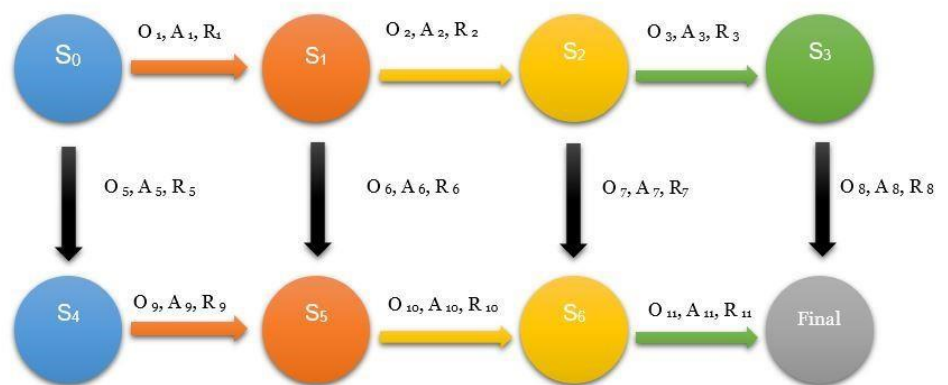
2.3.2.2. Enhanced OAI Model

The original OAI model identifies the possible actions in a system by representing each action in the form of an interface object. The connection between the action and the object is (1 to 1) correspondent. Enhanced OAI model is the same as the original OAI but it adds interface response. The OAI model is a state machine and can only switch state when the condition is met [9].

Interface Object (O): GUI is built up with e.g. buttons and dialogue boxes. That is a representation of action, what could change the system state. Interface objects must have virtual affordance, design constraints and natural mapping. Virtual affordance is for the user to recognise the object to do a specific task, e.g. the handle on a teacup represent obvious affordance for holding. A design constraint is to limit the user's action for an interface object. Natural mapping is for the user to understand the structure related to an object [9].

Interface Action (A): Interface action is associated with an interface object, and is the user's conceptual model of an interface object [9]. Interface actions are also decomposable into lower-level actions, e.g. saving a file, there are many steps involved including setting a file name and then writing it to a memory [10].

Interface Responded (R): Is the response associated with each interface object. The interface object is representing the behaviour of an interface action and the representations of that action are interface respond [9].



A = interface action, R = interface responded, O = interface object

Figure 4 State machine for enhanced OAI model

This state machine works as follow: the system is beginning at S_0 . A change of state happens when the system performs a type of action (A_x), on a specific type of object (O_x), and the respond R_x indicates a new state. To enhance the usability, the GUI must be designed with natural mapping, virtually constrained and affordance to clarify which symbol is associated with the right action [9].

2.3.2.3. NGOMSL

Natural GOMS Language is a part of the GOMS family and is based on cognitive complexity theory (CCT). It is suitable for practical applications, and it contains an explicit procedure for developing a GOMS model. Because the NGOMSL is a program form, it opens up for the possibility to make the method structure very clear and can represent general methods [11].

The technique for constructing NGMSL model is a top-down method. The top-level goals for the user are transformed into methods, and the methods can transform into new methods again and so on, until the methods only contain the operators e.g. button, moving the cursor [11]. Every goal and sub-goals are constructed up by pseudocode [12].

The procedure of using NGOMSL is as follow:

- **Step A:** Define the top-level goals that the user will accomplish
- **Step B:** Do the following steps:
 - B1: Draft a method to accomplish each goal with no more than 5 steps in a method. There would also not be more than one high-level goal in a step.
 - B2: Verify the methods, to check their length and the level of detail.
 - B3: If needed, go to a lower level of analysis by changing the high-level operators to accomplish goal operators and the provided method for corresponding goals.
- Finally, verify that all the operators in a method are primitives, if not examine if there could be a method for performing it.
- **Step C:** Document and check the analysis
- **Step D:** Check sensitivity to judgement calls and assumptions [12]

Because NGOMSL is based on CCT, it can be used to estimate the time it takes for the user to learn the how to use it, and the estimated time it takes for execution of a task [11].

In the NGOMSL model, it is possible to estimate the time a person requires to learn how to use the GUI. The estimation is based on the numbers of NGOMSL statements and a constant called “*Learning time parameter*” where the value is 30 seconds for rigorous procedure training or the constant can be 17 seconds for a typical learning situation [12].

$$\text{Pure Method Learning Time} = \text{Learning Time Parameter} * \text{Number of NGOMSL statements to be learned}$$

Formula 1 NGOMSL formula to estimate learning time [12]

By using NGOMSL it is possible to estimate the time it takes to memorise the information in the long-term memory. It can be accomplished by using the Model Human Processor parameters, where it takes 10 sec/chunk to save into the LTM. These chunks represent how complex a method is [12].

Execution time can be used to predict the time it takes to execute a method. The estimation can only be done on a specific task because it depends on the number of steps and operations can be determined [12].

$$\text{Execution Time} = \text{NGOMSL statement time} + \text{Primitiv External Operator Time} + (\text{Analyst – defined Mental Operator Time}) + \text{Wating Time}$$

Formula 2 NGOMSL formula to estimate execution time [12]

- NGOMSL Statement Time: Number of statements executed * 0,1 sec.
- Primitive External Operator: Total of times for primitive external operators.
- Analyst-Defined Mental Operator Time: Total of times for mental operators defined by the analyst.
- Waiting Time = Total time when the user is idle while waiting for the system [12].

2.3.2.4. Fitts's Law

Fitts's Law is a model used to predict the time it requires to move to point at a target, based on the object's size and the distance. The law was developed to find a model that could describe the relationship between the speed and accuracy moving towards a target on a display. It has shown that the Fitts's law has been useful for deciding where to locate the digital objects on the screen, what size it should be and distances between different objects [13].

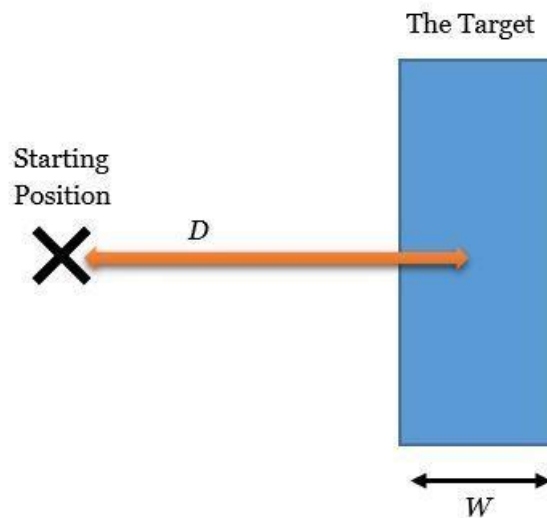


Figure 5 Fitts's Law

Fitts's Law equation is as follow:

$$ID = \log_2\left(\frac{D}{W} + 1\right)$$

Formula 3 Formula to calculate the index of difficulty [14]

$$T = k * \log_2\left(\frac{D}{W} + 1\right)$$

Formula 4 Formula for Fitts's Law, to calculate the time [13]

- ID = index of difficulty is measured in *bits* and is the difficulty of the task moment.
- T = average time to complete the movement in milliseconds
- k = Is a constant different dependent on the system, and most of the time approximately 200 ms/bit [13].
- W = Wide of the object.
- D = the distance between the starting point and the centre of the object [14].

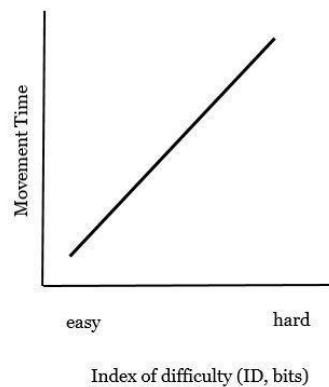


Figure 6 Movement time prediction [14]

Fitts's Law has been found useful to decide where to locate objects on the screen and its relation to other objects. The law also predicts that the 4 corners of the screen are the places which are the quickest to access. Fitts's law was used to help the designer predict the position and the sizes of the 12-keycell phone keypad [13].

2.3.3. Important design aspects

Every product's functionality is accomplished via HCI. The goal is to make the interaction between the person and product easier, to achieve higher usability. Especially if it is a two-way communication between the user and the computer that contain several processes: e.g. mutual transmission and feedback of information and mutual understanding [3].

2.3.3.1. Analysis of the users

The user of the system plays an important role in the development of the system. At the initial stage of system development, an evaluation of the users' needs in HCI must be done. The investigation of the potential user must include analysis of the users' skills and experience, to predicted how the users will reply to different designs of interaction. It is also important to investigate the hardware and software environment for the HCI in order to enhance the feasibility and ease of using the system [3].

Based on the analysis of the users and system environment, the most suitable type of HCI interaction should be formulated. This means to identify the task mode of HCI, estimating the level of support to the interaction and predicting the complexity of the interaction [3].

2.3.3.2. **Ergonomics**

By creating an ergonomic GUI and HCI the workload of the user will be lessened. The more the system and computer can do, the less workload for the user and the design will be better. Using a combination of humans and computers will effectively ensure longer life and better reliability for the system [3].

2.3.3.3. **Reducing the risk of user mistakes**

To achieve a GUI where the user makes as few mistakes as possible, it is important to make it understandable. By using similar interface appearance, layout, mode of interaction and information display, it will reduce the burden of the user. Because the user is already familiar with the design and the reaction, the user will not have to learn and memorise a new design [3].

2.3.3.4. **Identifiable and operational design**

The design of the interface should be simple and understandable, and the using of colour is the best effect of showing. Other ways to conveying information are the use of icons, graphs, language e.g. menus, windows, recycle bin, documents, folders, tool box. It also needs to be designed without any culture and religious barriers, to make the interface universal [3].

2.3.3.5. **Communicating between user and computer**

If a user by mistake is manipulating the system, the system should inform the user. Or to prevent an incorrect input the can make confirmations from the user [3].

2.3.3.6. **Shortcuts**

For the user in an easy and fast way to find information in a large amount of data a search function, or by using a scroll bar window. Other shortcuts should be provided to the user, based on the user's different experience or motive [3].

2.3.3.7. **Feedback**

Then the user is using the interface the user should have a respond to the operations. The feedback can be presented in different ways e.g. texts, graphics and sound [3].

2.3.3.8. **Short term memory**

Short term memory is a kind of memory that can be referred to as a scratch pad, to save temporarily information. The memory can be access to rapidly, at 70ms, but the saved information can only be kept in 200ms. Then remembering a phone number or a sequence of the number you usually remember it in parts, the average person can only remember 7 ± 2 digits. This aspect is imported in the designing of the system [6].

2.3.3.9. **Help function**

For the user to be able to find the GUI easy and understandable and effective, the interface should provide simple and standard help operation. All the operations based on HCI should have a help function. The help function and the content should be based on the knowledge of the user, and by using understandable terms and language to provide the content useful [3].

2.4. **Virtual design**

Virtual design is a purpose to make the design user-friendly and understandable for the user. The design is based on the psychology of light and the sensory organ of the target group of users. Virtual design includes e.g. the choice of colour, a method to present graphics and image, front design, page layout. In the following list are some principal of virtual design are stated [3]:

- *With the clear and coherent interface, it allows users to customise the contents of the interface.*
- *To enhance computer's function of memorising and reduce user's burden of short term memory.*
- *To provide more functions as default, undo and redo.*
- *To provide more interface shortcuts.*
- *Icon design should respect the past using the experience of the user.*
- *To enhance visual stimulants of graphics symbol through the application of colour. Different colours give us different feeling, check it on table 2.*
- *To improve the clarity of the visual symbol and make the picture, layout of words and metaphors easy to understand and identify.*
- *Make the whole colour of interface within five colour systems and minimise the use of red and green. Similar colours should be used in icons on behalf of similar meanings. [3]*

Table 2 Psychological reaction of colours [3]

Colour	Psychological reaction
Red	Fervency, peril, spark...
Orange	Warm, joy, envy ...
Yellow	Sunshine, hope, cheerfulness, commonness...
Green	Pace, safety, growth, greenness ...
Blue	Equability, mind ...
Purple	Elegance, dignity, weightiness, mystery
Black	Solemnity, vigorous, fear, death ...
White	Immaculate, holiness, lustration, sunshine...
Grey	Commonness, chill, modesty...

2.4.1. Light

Brightness is the amount of the light reflected from an object, and the reflection is measured by the amount of luminance. The brightness in displays that are used today is also measured in luminance. Using a display with high luminance will increase the visual activity of the user. The negative part is that using higher luminance will also increase the amount of flicker. It will only appear if the screen is switched less than 50 Hz if it is higher the eye will think it is on [6].

2.4.2. Military symbols

To get a global language of symbols that could be used for people from different countries and culture. NATO created its own symbol language, to be certain that soldiers from different countries would not misunderstand its meaning [15]. The following sections are describing how they are created.

2.4.2.1. Colour

In military symbols, the colour the object has is inside or at the boundary has different meanings, see table 3 [15].







Table 3 Colour specification from MIL-STD-2525D [15]

Description	Hand-Drawn	Computer Generated (ICON)	(FILL)
Friend	Blue	Cyan RGB:(0,255,255)	Crystal Blue RGB:(128,224,255)
Neutral	Green	Neon Green RGB:(0,255,0)	Bamboo Green RGB:(170,255,170)
Hostile	RED	RED RGB: (255,0,0)	Salmon RGB: (255,128,128)

2.4.2.2. Identities and dimensions

Dimensions and identities symbol is representing a mission area for the object within the operational environment. An object can have a mission area in air, land and sea or booth under and over the earth's surface. To make it easy all the different objects with is representing a mission is represented with its own dimension, e.g. following objects [15].

Table 4 Examples of dimensions for some area and objects [15]

Affiliation Dimension / Battle	Sea Surface	Air
Friend		
Hostile		
Neutral		

2.4.2.3. Icon

The icon is the placed at the innermost part in a symbol, which represents a unit, equipment, installation, activity and operation. An example of an icon is the representation of Medical, see figure 7 [15]:

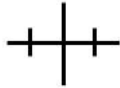
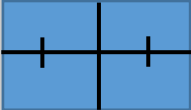

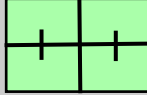


Figure 7 Icon for medical treatment facility

Table 5 Example of full frame icon [15]

Friendly	Hostile	Neutral
		
Medical treatment facility	Medical treatment facility	Medical treatment facility

2.4.2.4. Modifiers

A modifier provides additional information about the icon e.g. unit, installation or activity. The modifiers are placed either at the top or bottom inside the frame, see figure 8 [15].



Figure 8 Example of modifiers, there the modifier is mountain. The figure stands for Mountain Infantry

3. Methods and result

In this chapter, methods and techniques used in designing the new HCI model and prototype of the GUI are explained.

3.1. New HCI model

The result from the study of different HCI models in chapter 2, is that neither of them has a method for preventing misconfiguration. These models also describe and analyse HCI in different areas e.g. Fitts's Law is about the time estimation of movement compared to NGOMSL which has the same functionality but has predefined values. To solve this problem a new HCI model should be created.

The idea is to use some of the listed HCI models in chapter 2, and create a new HCI model based on these. The OAI model is based on a state machine and cannot move to the next state until the previous state is done. That can be used to be certain that the user has configured the system in a correctly. NGOMSL is a good way to divide the goals for the system to higher level goals and sub-goals. The idea is that the higher-level goals are representing a state machine and the sub-goals are representing different states for the state machine e.g. A high-level goal could be that the user must set 2 settings, each of these sub-goals is representing a state in the state machine. The user must configure these two settings before the high-level goal is accomplished. Fitts's law can be useful to estimate the time it takes for the user to move and for determining the size of the symbol. NGOMSL together with Fitts's law gives an estimation of the time it takes to perform the states, and where things should be placed on the screen. The design process can be divided into 3 steps: Task and goal, Structure, Design.

3.1.1. Task and goal

Defining the Task and goal of the system is done by following the NGOMSL model to define the high-level goal and then dividing it to sub-goals. The NGOMSL script and formula 1 and 2 in section 2.3.2.3 can together make an estimation of the time it takes to execute and learn each method, to optimise the high-level goal.

3.1.2. Structure

The sub-goals that are defined in the previous step is representing its own state and each state is defined by a *Symbol, Action and Response*. In the previous step there is no clarification for the symbols, the associated action, and response of accomplished goal. Then the system/GUI will be created, and there is a good overview structure of the method for programming the high-level goal.

- **Interface Symbol (O):** Is a symbol that is associated with a special type of action e.g. NGOMSL command, or it can be a symbol that provides data e.g. picture. The symbol is a part of the GUI e.g. a button. The design of the symbol must have virtual affordance to provide looks and feel requirements for a specific type and action, to reduce the risk if the user thinking that the symbol is associated with another action.
- **Interface Action (A):** is the operation done by the interface symbol applied by the operator. The action is applied to the system. In some action, it can also be sub-actions e.g. the user exit's the system and a sub-goal is to save the work the user has done before closing the program.
- **Interface Respond (R):** is the response from the state machine when it has changed its state from its previous state to a new state, and sent a response of that to the user.

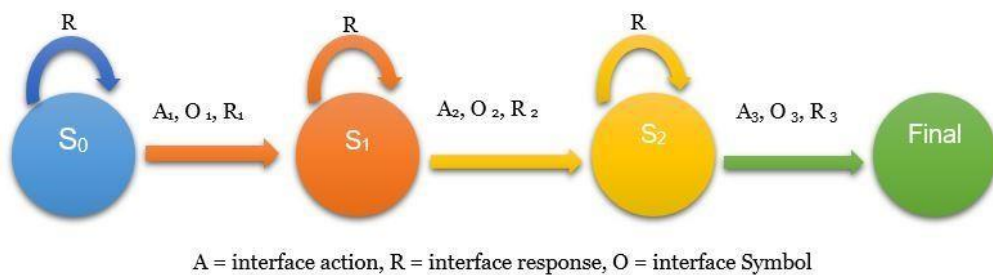


Figure 9 Example of state machine from start to the goal of a task

3.1.3. Design

The design is one of the most important aspects of the HCI because it is where the user will work and it must therefore be designed to minimise the workload of the user. This can be accomplished by placing the symbols in natural mapping to reduce the risk of the user thinking that the symbol is associated with the wrong object. Fitts's law is used for analysing the time it takes for the user to move the cursor from the previous position to the finish position, based on the size of the symbol and the distance from the starting point and the finish point. To fulfil all these aspects, the design of the GUI must consider the different aspects that are described in section 2.3.3 and 2.3.4.

3.2. **Solution methodology**

There are many ways to test the new HCI model to see if it can be used in a real scenario. The critical aspects of analysing are if the three steps in the model are enough or if it needs more steps, or perhaps if the steps need to be redefined to make them greater. To get some useful data from analysing the HCI model, there are three different ways, and they are as follow:

- The first method is to verify the new HCI model for a scientist in HCI and let him give feedback at what parts that need to be redefined and what parts to keep.
- The second one is to photoshop the finishing design of a GUI based on the HCI model. Then, a group of potential users will decide if the design is easy enough, or which parts that need to be redefined.
- The third one is to make a prototype of a GUI based on the new HCI model which will be created with all the three developing steps. The prototype will be evaluated by having a user provide feedback after performing a test from a pre-defined script.

The choice to continue this work is to go forward with the third choice. Because it is the only method that what will simulate the HCI model in a real development and allows getting feedback from a potential user.

3.3. **Prototype of the GUI with the new HCI model**

This section is about the prototyping of the GUI based on the new HCI model. The following sections are divided into different high-level goals, which are divided into the following three sections Task and goal, Structure and design. UML for the GUI is in appendix A.

In the following steps, the learning time parameter will be 30 seconds when estimating the learning time using formula 1.

3.3.1. Method to calculate the Fitts's law

Calculating the Fitts's law, described in section 2.3.2.4, requires two different values. The first is the distance between the start position and the end position. The second is the size of the target. Because Fitts's law do not require the values to be in the specific metric unit, the grid net of the screen can be used to calculate the distance. To get the distance from the grid, Pythagoras' theorem can be used by knowing the distance in X-axis and Y-axis between the two points.

A program was created to print out the coordinates by clicking at the desired point. To see the code, see appendix B.

3.3.2. Planning mode

The planning mode is the mode of the GUI where the user in advance can plan which missions who has the highest and lowest priority in the network. It will also be able to set the conditions of the nodes and communication types in a mission. The planning will be divided into which day the mission will be performed.

3.3.3. Set priority for the missions

The high-level goal of this step is to choose which priority the different missions has, it will be done by specifying which mission who has the highest and then rank them down to the lowest.

3.3.3.1. Task and goal

The high-level goal of this step is that the user will choose the priority between the different missions. The user chooses from a pre-defined list of numbers, and then set the mission with the highest priority to "1" and the lowest with the lowest number from the list. The range of numbers is equal to the numbers of missions e.g. if there are five missions, the list will be represented from 1 to 5. Number 1 is representing the highest priority and 5 the lowest.

The missions are divided into which day they will be performed and the user must choose which day he wants to plan for. From the start, the GUI will be defined to show the missions that will be performed at that day.

Goal: Set rank to the desired mission

Step 1: Decide: if no more tasks, then return goal accomplishment

Step 2: Accomplish goal: Choose day

Step 3: Accomplish goal: Move to location

Step 4: Accomplish goal: Set desired rank in the mission

Step 5: goto 1

Selection rule set for goal: Choose day

- If the task is plan for today, then accomplish goal: Move to location
- If the task is planned, then accomplish goal: Choose day

Goal: Choose day

- Step 1:** Locate the symbol represented with a calendar
- Step 2:** Move cursor to location of the symbol
- Step 3:** Click at the calendar
- Step 4:** Move cursor to the desired date
- Step 5:** Click at the desired date
- Step 6:** Return accomplishment

Goal: Move to location

- Step 1:** Locate the desired mission
- Step 2:** Move cursor to the location of the mission
- Step 3:** Click at the list of ranks
- Step 4:** Return accomplishment

Goal: Set desired rank on the mission

- Step 1:** Locate the desired rank to the mission from the list
- Step 2:** Move cursor on the list to the desired rank
- Step 3:** Click at the desired rank
- Step 4:** Return accomplishment

Figure 10 NGOMSL script for goal "Set priority for the missions"

From the formula 1 and 2 in 2.3.2.3 an estimation of time to learn and execute the goals can be done.

• **Estimation of the time to learn for each method:**

- *Choice of day:* 2,5 min
- *Move to location:* 1,5 min
- *Set desired rank on the mission:* 1,5 min

• **Estimation of time to perform each method:**

- *Choice of day:* 3,7 s
- *Move to location:* 2,15 s
- *Set desired rank on the mission:* 2,15 s

3.3.3.2. **Structure**

The following steps is the description of how the different sub-goals are represented as symbols for the user, and what the system will do the then action is activated, and the response it gives the it is executed

- **Interface Symbol:**
 - **Move to location:** This method is represented with a table view, the missions are represented in its own row. The row contains an overview of the mission, that informs the user about the name, rank, organisation and a short definition of the mission. The column that represents the rank is represented with a choice box to choose the desired rank.
 - **Set desired rank on the mission:** This method is represented by a list of numbers that represents the rank. The lowest value represents the highest priority and the lowest value is the one with lowest priority.
 - **Choose day:** This method is represented with a calendar, to choose which day the user wants to plan for.

- **Interface Action:**
 - **Move to location:** This methods action is that it will display the possible numbers of rank.
 - **Set desired rank for the mission:** Save the desired rank for the desired mission.
 - **Choose day:** The system will search for missions that will be performed that day.

- **Interface Response:**
 - **Move to location:** The response is that it will display the chosen rank will be displayed in the rank column.
 - **Set desired rank on the mission:** The response is that the front text of the list will change its value.
 - **Choose day:** The response is that the missions that were found at the desired day will be presented in the rows in the table view.

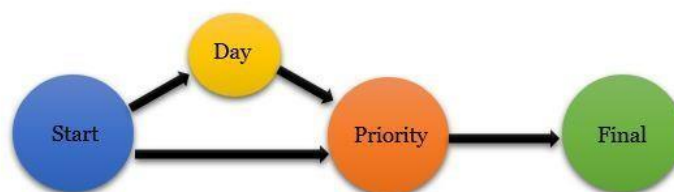


Figure 11 State machine for set priority for missions

3.3.3.3. Design

The result of this goal is as shown in figure 12. All the missions that will be performed the desired date will be listed in its own row and the user can see a short introduction about the mission and which organisation it belongs to. To select which mission who has the highest priority or the lowest, the user can double click at the rank column at each mission row and choose desired rank.

To reduce the working memory for what the different types of symbols does, and to give information about them, some of the symbols have been equipped with a tooltip. The symbols that represents the choice of date and the list of missions are equipped with a tooltip, see figure 13 and 14.

Mission	Rank	Orginasation	Mission info
Defend the hill	1	7 Bataljonen	Defend the hill from being ...
Defend the the food reserve	2	Livgardet	Defend the food from una...
Medivac	3	Skyttebataljonen	Evacuate injured civilians fr...
Standard operation	4	Livbataljonen	Defend The Royal Palace
Defend Musköbasen	5	Amfibiebataljonen	Defend Musköbasen from ...

Figure 12 Scene of the list with mission that will be performed at that day

Mission	Rank	Orginasation	Mission info
Defend the hill	1	7 Bataljonen	Defend the hill from being ...
Defend the the food reserve	2	Livgardet	Defend the food from una...
Medivac	3	Skyttebataljonen	Evacuate injured civilians fr...

Figure 13 The tooltip responds from the list of missions

Figure 14 The tooltip responds from the choice of date

Fitts's Law equation:

The following sections is the Fitts's law equations for the different symbols.

- **Choose Day:**

The starting point of the cursor will be estimated at the centre point of the screen. This equation is about the time it takes for the user to move from the start point to the date window. The distance was measured to 138 unit of length, and the size of the target is 200 units of length.

$$ID = \log_2 \left(\frac{138}{200} + 1 \right) = 0,76, \quad T = 152 \text{ ms}$$

Equation 1 Fitts's law equation for the sub-goal choose day

- **Move to location:**

This step is divided into two sections. The user comes either from the date window or it comes from the start point in the centre of the screen. The calculation will be to the first mission on the table view.

- o **Centre point (Start point):**

The distance from the start point and the first mission is 112 units of length and the width of the target is 100 units of length.

$$ID = \log_2 \left(\frac{112}{100} + 1 \right) = 1, \quad T = 200 \text{ ms}$$

Equation 2 Fitts's law equation for the sub-goal move to location

- o **Date window:**

The distance from the date window to the first mission was measured to 81 units of length and the width of the target is 100 units of length.

$$ID = \log_2 \left(\frac{81}{100} + 1 \right) = 0,856, \quad T = 171 \text{ ms}$$

Equation 3 Fitts's law equation for the sub-goal move to location

- **Select rank for the desired mission:**

The distance from the point where the user clicks at the list, to the first number is 29 units of length and the width of the target is 100 units of length.

$$ID = \log_2 \left(\frac{29}{100} + 1 \right) = 0,37, \quad T = 73 \text{ ms}$$

Equation 4 Fitts's law equation for sub-goal move to location

3.3.4. Edit properties for the desired mission

This section is about the HCI model for the high-level goal of *Edit properties for the desired mission*.

3.3.4.1. Task and goal

The goal of this task is that the user should be able to change the properties of a mission by changing the priority and quality for different types of communications types and nodes (objects) that are a part of the mission.

Goal: Edit Settings for Task

- Step 1:** Decide: if no more tasks, then return goal accomplishment
- Step 2:** Accomplish goal: Select desired mission
- Step 3:** Accomplish goal: Move to unit task location
- Step 4:** Accomplish goal: perform unit task
- Step 5:** goto 1

Selection role set for goal: move to edit one object

- If the task is to search for object to set priority and quality, then accomplish goal: search for object
- If the task is to find the object in a list, then accomplish goal: locate object from a list

Goal: Select desired mission

- Step 1:** Locate the desired mission from the list
- Step 2:** Move cursor to the desired mission
- Step 3:** Click at the desired mission
- Step 4:** Return with accomplishment

Goal: Search for object

- Step 1:** Locate the search window
- Step 2:** move cursor search window
- Step 3:** keystroke the two first letters in the name of the object
- Step 4:** keystroke the arrow key down
- Step 5:** keystroke enter
- Step 6:** Return with accomplishment

Goal: Locate object from a list

- Step 1:** Locate the area of the object
- Step 2:** Move cursor to the area of the object
- Step 3:** Click at the name of the are the object belongs to
- Step 4:** Locate the desired object
- Step 5:** Move cursor to the desired object
- Step 6:** Click at the desired object
- Step 7:** Return with accomplishment

- Goal:** Set priority and quality
- Step 1:** Move cursor to the levels of priority
- Step 2:** Click at the desired level of priority
- Step 3:** Move cursor to the levels of quality
- Step 4:** Click at the desired level of quality
- Step 5:** Move cursor to ok button
- Step 6:** Click at the ok Button

Figure 15 NGOMSL script for edit properties for the desired mission

From the formula 1 and 2 in 2.3.2.3 an estimation of time to learn and execute the goals can be done.

- **Estimation of the time to perform each method:**

- *Search for object: 2,77 seconds*
- *Select desired mission: 2,15 seconds*
- *Locate object from a list: 4,3 seconds*
- *Set priority and quality: 4,4 seconds*

- **Estimation of time to learn for each method:**

- *Search for object: 2,5 min*
- *Select desired mission: 1,5 min*
- *Locate object from a list: 3 min*
- *Set priority and quality: 3 min*

3.3.4.2. Structure

The following steps is the description of how the different sub-goals are represented as symbols for the user, and what the system will do then the action is activated, and the response it gives then its executed

- **Interface Symbol:**

- **Search for object:** This sub-goal is represented by search window, to search for objects.
- **Select desired mission:** This sub-goal is represented with a table view that contains the different missions that will be performed at the desired date.
- **Locate object from a list:** This sub-goal is represented as tree view with areas and sub-areas e.g. an area can be tracking and the sub-area is BFT.
- **Set priority and quality:** This sub-goal is represented with a button and two combo boxes to set priority and quality.

- **Interface action:**
 - **Search for object:** Search for objects that contain the two typed in letters, and collect the information for the desired object.
 - **Select desired mission:** Collects the information that the desired mission contains of e.g. from nodes and interfaces.
 - **Locate object from a list:** Collect the information required to set priority and quality for the desired.
 - **Set priority and quality:** Save the desired priority and quality for the desired object.

- **Interface Response:**
 - **Search for object:** Presenting objects that contain the typed two letters and a window that contains the goal *Set priority and quality* goal will be displayed for the desired object
 - **Select desired mission:** The response is that it will open a new window that is an overview of the mission, that contains information about start and end date/time.
 - **Locate object from a list:** Presenting the desired subareas and a window that contains the goal *Set priority and quality* goal will be displayed for the desired object
 - **Set priority and quality:** if the user has configured the sub-goal correctly and then clicks at the ok button, it will close the window, otherwise it will remain open until the user has configured it correctly.

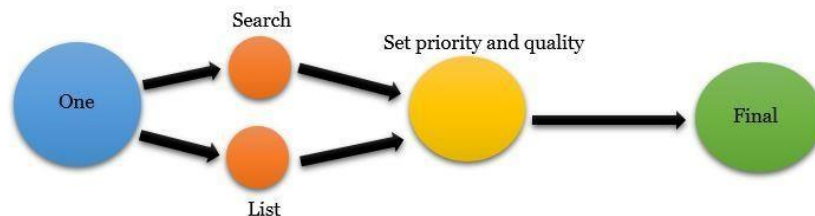


Figure 16 State machine for the goal “Edit properties for the desired mission”

3.3.4.3. Design

To choose which mission to edit the settings for, the user clicks at the row that belongs to the desired mission, see figure 12.

The user chooses a mission, a new window will open where the user gets an overview of the mission. The overview scene is about presenting the basic information about the mission. There is a short task definition, then information when the mission will be performed as well as when it ends, see figure 17.

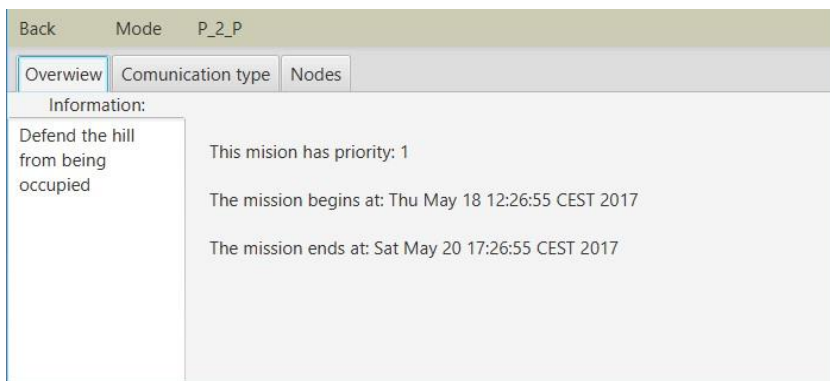


Figure 17 Overview scene for the chosen mission

To change the settings for the different communication types, the user opens the window "Communication type". There are two ways of opening the desired communication type, the first way is locating it from the different areas and the second by searching from the search window, see figure 18. The tab for nodes has the same functionality as this, see figure 19.

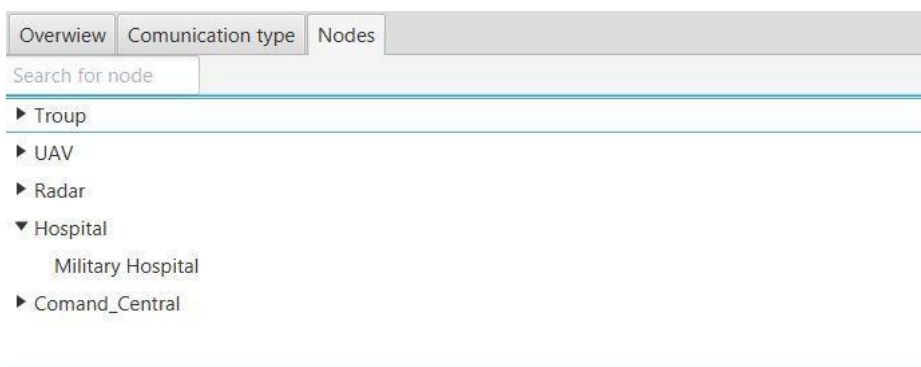


Figure 18 The window there all the different nodes types that belongs to the mission is presented

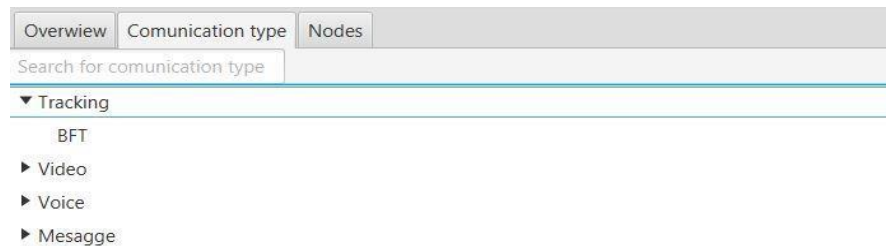


Figure 19 The window there all the different communications types that belongs to the mission are presented

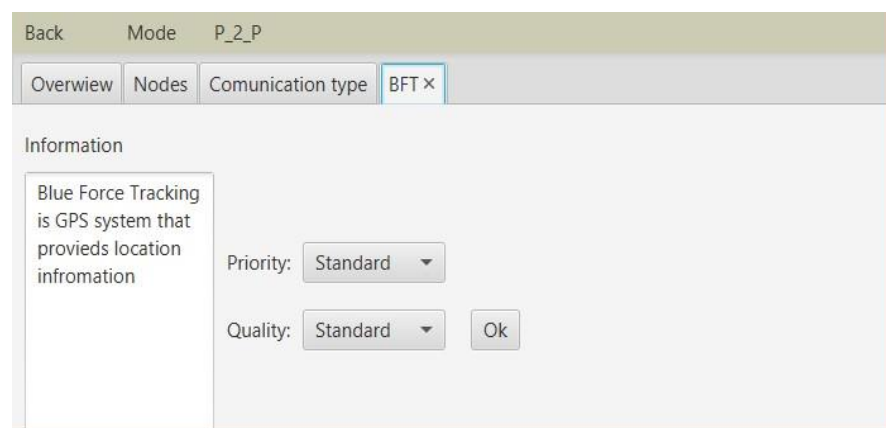


Figure 20 The screen to change the properties of communication types.

The screen in figure 20 is where the user can change the properties of the node or the communication type. To reduce the workload of the user, the screen contains only a small amount of information about the node or the communication type, and for nodes there is a military symbol describing the node, see figure 21. But still gives enough information to ensure the user that he or she has chosen the right one.

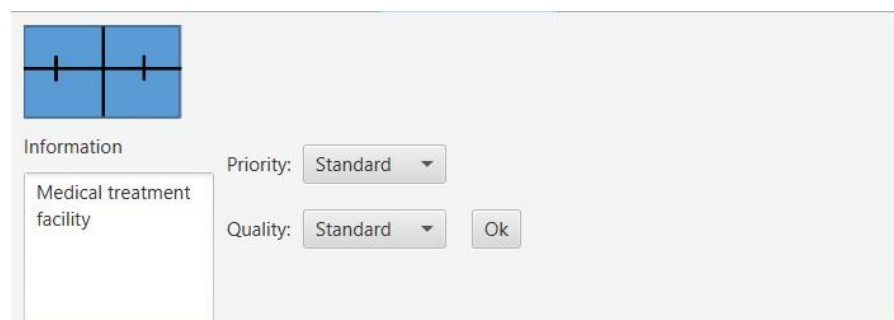


Figure 21 The screen to change the properties of nodes

Fitts's Law equation:

The following sections are the Fitts's law equations for the different symbols.

- **Select Mission:**

The user can select the mission that is listed as figure 12. One way is from date window or from the start point that is the middle point of the screen. This following calculation is for the first listed mission in the table view.

- **From calendar window:**

The distance from the calendar window to the first row of the mission was measured to 81 units of length and the width of the target was 700 units of length.

$$ID = \log_2 \left(\frac{81}{700} + 1 \right) = 0,16, \quad T = 19 \text{ ms}$$

Equation 5 Fitts's law equation for sub-goal select mission

- **From Start Point:**

The distance from the middle point to the first row was measured to 47 units of length and the width of the target is 700 units of length.

$$ID = \log_2 \left(\frac{47}{700} + 1 \right) = 0,094, \quad T = 18,8 \text{ ms}$$

Equation 6 Fitts's law equation for the sub-goal select mission

- **Select tabs:**

The estimated place there the user will select the mission is from clicking at the name of the mission. The following calculation is for the first

- **Node:**

The distance from point of the screen the user clicks at the first mission to the tab named node was measured to 63 units of length and the target has a width of 52 units of length.

$$ID = \log_2 \left(\frac{63}{52} + 1 \right) = 1,1, \quad T = 229 \text{ ms}$$

Equation 7 Fitts's law equation for the sub-goal select tabs

- **Communication type:**

The distance from point of the screen the user clicks at the first mission to the tab named node was measured to 87 units of length and the target has a width of 135 units of length.

$$ID = \log_2 \left(\frac{87}{135} + 1 \right) = 0,72 , \quad T = 143 \text{ ms}$$

Equation 8 Fitts's law equation for the sub-goal select tabs

- **Select item from a tree list:**

The equation of the two following steps is only for the first area and its first sub-area since the width of the area and its subarea depend on how long the word is.

- **The first area of communication type:**

The distance from the start point, which is the place in the tab the user clicks to open it, to the first area. Was measured to 99 units of length and the width of the target is 150 units of length.

$$ID = \log_2 \left(\frac{99}{150} + 1 \right) = 0,73 , \quad T = 146 \text{ ms}$$

Equation 9 Fitts's law equation for the sub-goal select item from a tree list

- **The first sub-area for the first area of communication type:**

The distance from the first area to the first sub-area was measured to 35 units of length and the width of the target to 37 units of length.

$$ID = \log_2 \left(\frac{35}{37} + 1 \right) = 1 , \quad T = 200 \text{ ms}$$

Equation 10 Fitts's law equation for the sub-goal select item from tree view

- **The first area of Nodes:**

The distance from the start point, which is the place in the tab the user clicks to open it, to the first area. Was measured to 99 units of length and the width of the target is 75 units of length.

$$ID = \log_2 \left(\frac{99}{75} + 1 \right) = 1,2 , \quad T = 242 \text{ ms}$$

Equation 11 Fitts's law equation for the sub-goal select item from a tree view

- **The first sub-area for the first area of nodes:**
The distance from the first area to the first sub-area was measured to 37 units of length and the width of the target to 75 units of length.

$$ID = \log_2 \left(\frac{37}{75} + 1 \right) = 0,58 , \quad T = 116 \text{ ms}$$

Equation 12 Fitts's law equation for the sub-goal select item from a tree view

- **Search window:**

The users also have the opportunity to search for an object, instead of finding it from the tree view.

- **Node:**
The distance from the point where the user clicks to open the tab to the search window was measured to 32 units of length and the width of the target is 150 units of length.

$$ID = \log_2 \left(\frac{32}{150} + 1 \right) = 0,28 , \quad T = 56 \text{ ms}$$

Equation 13 Fitts's law equation for the sub-goal search window

- **Communication type:**
The distance from the point where the user clicks to open the tab to the search window. Was measured to 28 units of length and the width of the target is 220 units of length.

$$ID = \log_2 \left(\frac{28}{220} + 1 \right) = 0,17 , \quad T = 34 \text{ ms}$$

Equation 14 Fitts's law equation for the sub-goal search window

3.3.5. Point to Point

This section is about how to set up a point to point connection between two nodes.

3.3.5.1. Task and goal

The Goal is that the user will configure a connection between two points in the network and set the priority and quality for the desired communication type.

Goal: Point to point connection

Step 1: Decide: if no more tasks, then return goal accomplishment

Step 2: Accomplish goal: Click the button

Step 3: Accomplish goal: Chose of Nodes

Step 4: goto 1

Goal: Click the button

Step 1: Locate the "P_2_P" button

Step 2: Move cursor to "P_2_P" button

Step 3: Click at the "P_2_P" button

Step 4: goto 2

Goal: Chose of Nodes

Step 1: Accomplish goal: Choose of node 1

Step 2: Accomplish goal: Choose of node 2

Step 3: Accomplish goal: Select type of communication type

Step 4: Accomplish goal: Set priority and quality

Step 5: Return with goal accomplishment

Goal: Choose of node 1

Step 1: Locate the search window

Step 2: Move cursor to search window

Step 3: Click at the search window

Step 4: Keystroke the two first letters in the node's name

Step 5: Keystroke the arrow down

Step 6: Keystroke the enter key

Step 7: return with goal accomplish

Goal: Choose of node 2

Step 1: Locate the search window

Step 2: Move cursor to search window

Step 3: Click at the search window

Step 4: Keystroke the two first letters in the node's name

Step 5: Keystroke the arrow down

Step 6: Keystroke the enter key

Step 7: return with goal accomplish

Goal: Select type of communication type

Step 1: Locate the search window

Step 2: Move cursor to search window

Step 3: Click at the search window

Step 4: Keystroke the two first letters in the node's name

Step 5: Keystroke the arrow down

Step 6: Keystroke the enter key

Step 7: return with goal accomplish

- Goal:** set priority and quality for one
- Step 1:** Move cursor to the levels of priority
- Step 2:** Click at the desired level of priority
- Step 3:** Move cursor to the levels of quality
- Step 4:** Click at the desired level of quality
- Step 5:** Move cursor to "ok" button
- Step 6:** Click at the ok Button
- Step 7:** Return with accomplishment

Figure 22 NGOMSL script to make a point to point connection

From the formula 1 and 2 in 2.3.2.3 an estimation of time to learn and execute the goals can be done.

- **Estimation of time to learn for each method:**

- *Click the button:* 1,5 min
- *Choose of object 1:* 3 min
- *Choose of object 2:* 3 min
- *Select type of communication:* 3 min
- *Set priority and quality:* 3 min

- **Estimation of the time to perform each method:**

- *Click the button:* 2,15 seconds
- *Choose of object 1:* 3,75 seconds
- *Choose of object 2:* 3,75 seconds
- *Select type of communication:* 3,57 seconds
- *Set priority and quality:* 4,5 seconds

3.3.5.2. Structure

The top goal is to make a point to point connection between two nodes and set which type of communication, priority and quality. To accomplish the top goal, there are some sub-goals which are described below:

- **Interface Symbol:**
 - **Click the button:** The method is represented by a button named P_2_P.
 - **Choose of object 1:** This method is represented with a search window and auto complete, and other symbols are the keys: enter, arrow down and the two first letters in the desired node's name.

- **Choose of object 2:** This method is represented with a search window and auto complete, and other symbols are the keys: enter, arrow down and the two first letters in the desired node's name.
- **Select type of communication:** This method is represented with a search window and auto complete, and other symbols are the keys: enter, arrow down and the two first letters in the desired node's name.
- **Set priority and quality:** This method is represented by two lists to present the priority respectively quality level.
- **Interface Action:**
 - **Click the button:** Open the window that contains all the other interface symbols.
 - **Choose of object 1:** Searching for objects that have the two first letters of the name similar, and auto-completion.
 - **Choose of object 2:** Searching for objects that have the two first letters of the name similar, and auto-completion.
 - **Select type of communication:** Searching for objects that have the two first letters of the name familiar, and autocompletion.
 - **Set priority and quality:** The system will save the desired priority and quality for the desired objects.
- **Interface Response:**
 - **Click the button:** Opens the window that contains all the other interface symbol.
 - **Choose of object 1:** Through auto completion presenting objects that have similar two first letters.
 - **Choose of object 2:** Through auto completion presenting objects that have similar two first letters.

- **Select type of communication:** Through auto completion presenting objects that have similar two first letters.
- **Set priority and quality:** Presenting the chosen level of priority and quality with a symbol.

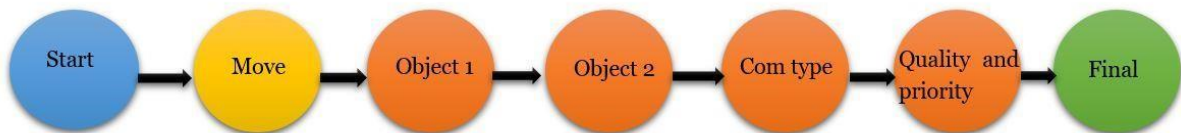


Figure 23 State machine for point to point communication

3.3.5.3. Design

The design of this goal came to be as shown in figure 24. The user can type in the two first letters of the desired node or communication type, and the system responds with nodes that contain the two letters, see figure 25.

Figure 24 The scene for Point to Point communication

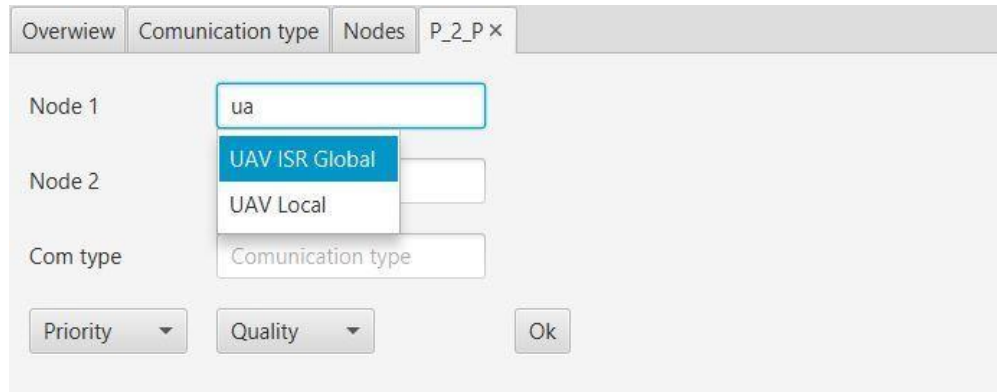


Figure 25 Example of auto completion

To reduce the working memory the different types of symbols, and the action they perform, the search boxes have been equipped with a tooltip which gives short information about each symbol. The reason is also to reduce the risk that the user is configuring this step in the wrong way, see figure 26,27 and 28.

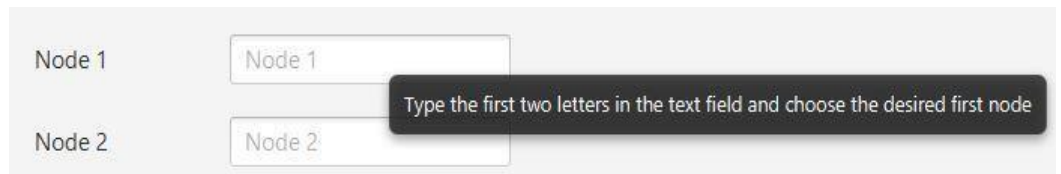


Figure 26 Tooltip for node 1



Figure 27 Tooltip for node 2

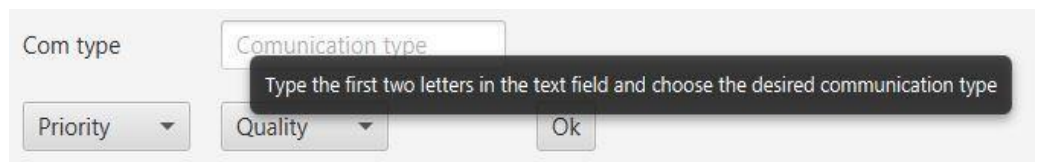


Figure 28 Tooltip for communication type

Fitts's law equations:

The following sections are the Fitts's law equations for the different symbols.

- **Search window Node 1:**

The start position is from the tab window, and the end point is the search window to node 1. The distance between the two points was measured to 66 units of length and the width of the target to 187 units of length.

$$ID = \log_2 \left(\frac{66}{187} + 1 \right) = 0,43 , \quad T = 87 \text{ ms}$$

Equation 15 Fitts's law equation for the sub-goal search window node 1

- **Search window Node 2:**

The start position is from the search window for node 1 and the end point is the search window for node 2. The distance was measured to 52 units of length and the target to 187 units of length.

$$ID = \log_2 \left(\frac{52}{187} + 1 \right) = 0,37 , \quad T = 74 \text{ ms}$$

Equation 16 Fitts's law equation for the sub-goal search window node 2

- **Search window Communication type:**

The start position is from the search window for node 2 and the end point is the search window for communication types. The distance was measured to 48 units of length and the target to 187 units of length.

$$ID = \log_2 \left(\frac{48}{187} + 1 \right) = 0,33 , \quad T = 66 \text{ ms}$$

Equation 17 Fitts's law equation for sub-goal search window for communication types

- **Choice of Priority:**

The start position is from the search window for communication types and the end point is the combo box to select a priority. The distance was measured to 173 units of length and the target to 113 units of length.

$$ID = \log_2 \left(\frac{173}{113} + 1 \right) = 1,34 , \quad T = 268 \text{ ms}$$

Equation 18 Fitts's law equation for sub-goal choice of priority

- **Choice of Quality:**

The start position is from the combo box to select priority and the end point is the combo box to select quality. The distance was measured to 53 units of length and the target to 113 units of length.

$$ID = \log_2 \left(\frac{53}{113} + 1 \right) = 0,556 , \quad T = 111 \text{ ms}$$

Equation 19 Fitts's law equations for sub-goal choice of quality

- **Ok button:**

The start position is from the combo box to select quality and the end point is the ok button, to close the tab. The distance was measured to 209 units of length and the target to 39 units of length.

$$ID = \log_2 \left(\frac{209}{39} + 1 \right) = 2,67 , \quad T = 534 \text{ ms}$$

Equation 20 Fitts's law equation for the sub-goal ok button

3.3.6. Live mode

At this mode, the user is able to see which nodes and communications types that are not operating at the desired conditions.

3.3.7. Select Mission

This section is about the goal *Select Mission*.

3.3.7.1. Task and goal

The aim of this step is to give the user an overview of the missions, that are executed at that specific time. From the GUI, the user will be able to see which nodes within each mission that are not operating at the desired conditions. To fix these problems, the user can reconfigure the settings so it can operate correctly.

The following figure 29 is the NGOMSL script for this goal.

Goal: Edit properties at nodes that is not operating at the desired level.

Step 1: Decide: if no more tasks, then return goal accomplishment

Step 2: Accomplish goal: Locate desired mission

Step 3: Accomplish goal: Edit properties

Step 4: goto 1

- Goal:** Locate desired mission
Step 1: Locate desired mission
Step 2: Move cursor node button at the desired mission
Step 3: Click at the node button
Step 4: Return accomplishment
- Goal:** Edit properties
Step 1: Accomplish goal: Select node
Step 2: Accomplish goal: Edit properties of the desired communication type
Step 3: Return accomplishment
- Goal:** Select node
Step 1: Locate the desired node
Step 2: Move cursor to the desired node
Step 3: Locate the desired communication type
Step 4: Move cursor to desired communication type
Step 5: Click at the desired communication type
Step 6: Return accomplishment
- Goal:** Edit properties of the desired communication type
Step 1: Move cursor to the levels of priority
Step 2: Click at the desired level of priority
Step 3: Move cursor to the levels of quality
Step 4: Click at the desired level of quality
Step 5: Move cursor to ok button
Step 6: Click at the ok Button
Step 7: Return with accomplishment

Figure 29 NGOMSL script for the goal select mission

From the formula 1 and 2 in 2.3.2.3 an estimation of time to learn and execute the goals can be done.

- **Estimation of time to learn for each method:**
 - *Locate desired mission:* 1,5 min
 - *Select node:* 1,5 min
 - *Edit properties of the desired communication type:* 3 min
- **Estimation of the time to perform each method:**
 - *Locate desired mission:* 2,05 seconds
 - *Select node:* 3,8 seconds
 - *Edit properties of the desired communication type:* 4,5 seconds

3.3.7.2. Structure

The following steps describe how the different sub-goals are represented as symbols for the user, and what the system will do the when the action is activated, as well as the response it gives when it is executed.

- **Interface Symbol:**
 - **Locate desired mission:** This method is represented as a table view where all the different mission is listed in its own row. The rows contains information about the mission and an indicator of how many nodes in the mission that are operating at the desired level (presented in percentages). To see which nodes that are not operating at the desired level, the user must click at the *node* button at the desired mission's row.
 - **Select node:** This method is represented with a list of nodes that are not operating at the desired level, and when the cursor is over the specific node, it will present the different communication types.
 - **Edit properties of the desired communication type:** This method is represented with a tab, and contains information about the communication type. To reconfigure the communication type the user has to change the priority and the quality, that are represented as combo boxes.
- **Interface Action:**
 - **Locate desired mission:** When the user clicks at the node button, the system must present which node that is not operating as desired. If the user moves the cursor over a node, the system presents the communication type that is not operating.
 - **Select node:** When the user moves the cursor over a node, the system must collect the communication types that is not operating as desired.
 - **Edit properties of the desired communication type:** When the user sets the priority, the system will save it temporality and look up the possibility to choose the quality. First when the user clicks at the ok button, the system executes the new settings.

- **Interface Response:**
 - **Locate desired mission:** The response to the user is that a list of different nodes will be presented.
 - **Select node:** When the user moves the cursor over a node, the user will be presented with the communication type that is not working at the desired level.
 - **Edit properties of the desired communication type:** The response when the user has set a value for the priority, the combo box for the quality will be looked up. Then the user clicks at the ok button, the tab will close if everything is configured.

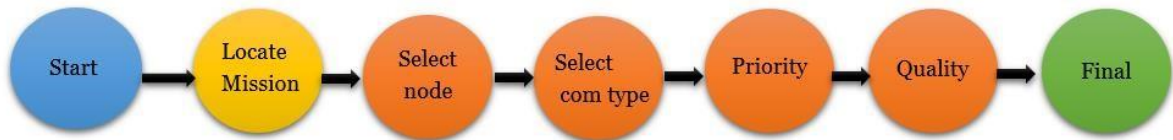


Figure 30 State machine for the goal Select mission

3.3.7.3. Design

The design of this step come to be as figure 31. From the overview, the user will see all the missions that are executed at that time. Each mission is represented by its own row that contains information about the mission and an indicator of how many nodes in the mission that is operating as desired. To reduce the workload of the by looking at all the mission for the operating condition the user can filter them so it will sort them all, from worst to best or the opposite. It can also be sorted by the name of the mission. Another feature to reduce the workload is that you can place the cursor over the information column at each row and the full text of information will be presented as a tooltip, see figure 33.

To edit the properties of communication types, the user must first select which the mission and the click at the node button at that row. To present the list of nodes, then the cursor is pointing at a node, a list of communication types that is not operating at the desired level will be displayed, see figure 32. The scene to change the settings of the desired communication type is the same as figure 21.

Back Mode P_2_P					
Live					
Select	Mission	Info	Operating nodes	Nodes with Error	
<input type="checkbox"/>	Defend the hill	Defend the hill fro...	100%	Nodes	
<input type="checkbox"/>	Medivac	Evacuate injured ci...	75%	Nodes	
<input type="checkbox"/>	Standard operation	Defend The Royal P...	75%	Nodes	
<input type="checkbox"/>	Defend Musköbasen	Defend Musköbase...	75%	Nodes	

Figure 31 The overview scene of live mode

Back Mode P_2_P					
Live					
Select	Mission	Info	Operating nodes	Nodes with Error	
<input type="checkbox"/>	Defend the hill	Defend the hill fro...	100%	Nodes	
<input type="checkbox"/>	Medivac	Evacuate injured ci...	75%	Nodes	
<input type="checkbox"/>	Standard operation	Defend The Royal P...	75%	Military Hospital	Contol
<input type="checkbox"/>	Defend Musköbasen	Defend Musköbase...	75%	BMS/Soldier	

Figure 32 The tooltip response then the user moves the cursor to the information column

<input type="checkbox"/>	Defend the hill	Defend the hill fro...	100%	Nodes	
<input type="checkbox"/>	Medivac	Evacuate injured ci...	75%	Nodes	

Figure 33 The scene of live mode to select communication type

Fitts's Law equations:

The following sections are the Fitts's law equations for the different symbols.

- **List of Nodes:**

The start point is the position the cursor is after the user has select to go to live mode. From that position to the node button the distance is 146 unit of length, and the width of the target is 90 unit of length. This calculation of time is for the first mission listed in the table.

$$ID = \log_2 \left(\frac{139}{90} + 1 \right) = 1,39 , \quad T = 278 \text{ ms}$$

Equation 21 Fitts's Law equation for the movement from the start point to the target

- **Select node:**

The start point is from the previous step's end point, and the target is the first node in the list, see figure 30. From the start point and to the target is a distance of 40 unit of length, and the target has a width of 150 unit of length. This calculation is only for the first target in the list.

$$ID = \log_2 \left(\frac{40}{150} + 1 \right) = 0,34 , \quad T = 68,2 \text{ ms}$$

Equation 22 Fitts's Law equation for the movement from the start point to the target

- **Select Communication type:**

The start point is from the selected node and endpoint is the listed communication type, see figure 30. From start to the end point is a distance of 139 unit of length, and the width is 70 unit of length. This calculation is for the first target in the list.

$$ID = \log_2 \left(\frac{139}{70} + 1 \right) = 1,58 , \quad T = 315 \text{ ms}$$

Equation 23 Fitts's Law equation for the movement from the start point to the target

3.3.8. Point to Point

This goal is the same as in the section 3.2.4.2, for more information about this goal, look at that section.

3.3.9. Simulation mode

The simulate mode has the same functionality as the live mode, but instead of presenting which communication type or nodes that are not receiving the required condition in real time. It will instead present the performance of the system, if the network can deliver the required conditions.

3.4. Test on potential user

The purpose of the test is to analyse if a GUI based on the new HCI model could reduce the risk of the system being misconfigured. The test was designed so that the user had to configure the system from the predefined script, see appendix C.

The potential user in this test is an employee at SAAB and has the same background knowledge as a supposable real user.

This test does not include the simulation mode because it has the same design and user response as the live mode.

The result of the potential user:

The user succeeds to set the desired settings from the predefined script, without any navigation problem or misunderstandings. The following section is the pros and cons the user thought about the different modes.

Planning mode:

- The user thought the design of this step was good and the information that is presented to get an overview is sufficient.
- The ranking system to set the priority of the mission was a good idea.
- The mapping and the size of objects were good and natural, and it was no goal or steps that were difficult to navigate around.
- The user did not feel that it was difficult to navigate around or to do some task, considering the new HCI model that different sub goals must be executed before completing a goal. Especially at P-2-P goal, their user must set the settings in a specific order.

Live/Simulation mode:

- The user thought the information provided to get an overview of the missions operating condition was sufficient.
- The user thought the indication of the percentage of working nodes should be indicated with a colour scheme, to get a quick overview of their operating condition.
- The mapping and the size of objects were natural and easy to locate.

The conclusion of this test is that the new HCI model is reducing the possibility to configure the system in wrong way and without incising its complexity.

4. Analysis and discussion

This chapter analyses and discusses the performed work and how the HCI model can be used for a sustainable development.

4.1. Choice of HCI model

The NGOMSL model were chosen because it uses a high-level goal that is divided into sub-goals. There the sub-goals are described as a pseudocode and can be evaluated from the time it takes to execute the goal and the learning time. The NGOMSL scripts create a structure of goals that needs to be accomplished to accomplish the high-level goal based on the sub-goals. That creates a structure of goals that can be used as a state machine with sub-goals that need to be done before the state machine can change its state.

Enhanced OAI model was chosen because it uses a state machine with sub-goals from NGOMSL script. The sub-goals are defined from which symbol the goal is associate with, the action it will have on the system and the response the user gets when the goal is accomplished.

The design step was the hardest to define because there are a lot of systems today that uses GUI to control them. Creating a method that could be used to all of them would be hard and complex. The solution was to use Fitts's law that lets the designer place the symbols as desired, as long the movement time and indexes of difficulty have a desired value.

4.2. New HCI model

When creating the new model, no consideration about the standards was taken account for. This resulted in that all the GUI that will be developed by this method will not fulfil the standards for human-computer interaction and graphical user interfaces such as ISO 9241.

None of the defined steps in the new HCI model is analysing the characteristics of the potential user or system. Therefore, a new step at the beginning of the system development (before starting with the other steps) should be to investigate the user from the aspects described in section 2.3.3.1.

4.2.1. Task and goal

From using the NGOSML method for describing the Task and goal, it was found to be useful creating a structure of how the user interacts with the GUI. It was also useful to minimise the NGOMSL goals to 5 steps and if needed, divide the problem into two goals. This created a good structure of what needed to be accomplished before starting the next goal and resulted in the creation of the state machine useful in the *Structure* step. From describing the goal in NGMSOL script it was found useful to analyse all the steps the user must go through to accomplish the goal. For an example, a goal can be to make the user type a word, it is preferred if the text box is already focused so the user do not need to move the mouse to the text box and the click at it. This saves the user both the movement time from the start point to the end point as well as the time it takes to click at the text box.

This step would be expanded with an action UML with all the actions the user must accomplish to reach the goal. During development of the prototype GUI, the NGOMSL script was showed to someone who does not understand it and that person got confused.

4.2.2. Structure

It was found useful to list all the goals in the categories *Symbol*, *Action* and *Response* because they schematically describe how the system and the user will see each goal. Describing what each symbol represents within each goal and what response the system will give when it is activated.

4.2.3. Design

This section is hard to create rules for because today the user does not only interact with the system by keyboard and mouse. Instead, touchscreens are getting more common and the design of a GUI for a touchscreen is different from the traditional one. The main aspect is the vast variety of different system types represented with a GUI and making a rule that fits all these systems are therefore very difficult. The only solution is to keep this step as it is, and use Fitts's law to calculate the index of difficulty and the moment time.

The negative aspect of Fitts's law is that there is no predefined interval. When a Fitts's law calculation is done, there are no definitions that describe if the value is good or bad. It is up to the designer to evaluate the value. GUI: s created from this model could therefore lead to the designs of different qualities from an HCI perspective.

4.3. **Solution methodology**

The test of the new HCI model included making a prototype of a GUI, that was tested on a potential user. To get more significance to this study, more extensive tests should have been done. One way would have been, to begin with, to present the HCI model to a scientist in HCI for evaluation if the model is even feasible. The next step would have been to interview people that would work at the different steps *Task and goal*, *Structure and Design* and let them evaluate the steps to make the them more structured and productive. The focus of this new model has only been for a user in the military and no other working groups, but it would also have been good to evaluate the model to other working groups e.g. heal care.

4.4. **Test of GUI**

The test with a potential user could have been bigger, for an example with a test panel with people of different age and culture. That would have led to a result that would have shown if there was any difference in how the user experienced the GUI depending on the user's age or cultural identity. One of the criteria in the HCI model and in section 2.3.3 was to develop a design which functionality would not be affected by the user's cultural identity.

The results of the test with a potential user at SAAB is that this method reduces the possibility of configuring the system in a wrong way. Even if the user had to configure the system from a predefined script, the user felt that the design was clear and it was easy to navigate around.

4.5. **Sustainable development**

One of the reasons to use theoretical HCI models to create a GUI is to reduce the developing time. Instead of designing a GUI and the let a panel evaluate the design, it can be created from a model that will result in a GUI that is easy to use and therefore reduces the workload. Reduced developing time also leads to reduced developing cost of the GUI. That could lead to decreased prices and the efficiency of the developing team would increase as they can take on more projects at a smaller time schedule.

A negative aspect of using HCI models could be that the GUI will be too easy to use. That could lead to the user getting bored in lack of stimulation from complex tasks.

Another aspect of using HCI models is that if the results turn out to be very good and easy to use, the customer doesn't need to buy newer versions of the product. Thereby the costumer can save money but in the same way, the company do not get to sell as many products of the newer version, which will have a negative effect on their economy.

If the systems that are used in critical situations e.g. medical care and industry where the systems need to be configured carefully and correct, safety can be increased by using this new HCI model. Better functioning systems could lead to improved quality of service from health care, by reducing the accidents made from a misunderstanding between the system and the user. In the industry, this could lead to lower manufacturing costs because the time it takes to control and regulate the developing process can be reused, and the accuracy can be increased.

If systems used in working environment are developed with HCI the workload of the user will be reduced. That could lead to a better working environment because it will be easier and faster to find things and perform tasks through the GUI. That could reduce the stress on the operator.

Human computer-interaction will be an important area in the upcoming years for the development of Internet of Things products. The interaction between the user and the computer will be increased by all the IoT products surrounding the user. This requires more research in to finding new interaction channels and improving the channels used today e.g. fragrance interaction for people that cannot hear or see, and make the voice interaction and understanding greater. To make a system that can be used by wider group of users and to make their life more intelligent and humane.

5. Conclusion

The goal of this was to evaluate different HCI models that could reduce the possibility that the system was configured in a wrong way by the user. For a GUI at SAAB, to regulate the priority and quality between the information provider and consumer. It resulted in that there are no HCI models that is taking that aspect into account, so a new model had to be created. The new model is divided into three steps *Task and goal*, *Structure and Design*. The first step is to create an NGOMSL script of the goals and sub-goals, where each goal can be evaluated on the learning time and execution time. Next steps are to take the sub-goals from the previous step and create a state machine from them that represent a higher goal. The user has to accomplish the goal before it can move to the next sub-goal. The last step is the design of the GUI, by using Fitts's law to decide the size and the place of the object.

The new HCI model was used to create a GUI for the project and then tested on a potential user. The results indicate that the HCI model reduces the risk of misconfiguration and that it was easy to use and navigate around in.

6. Continuous work

The new HCI model that was develop was only tested for a GUI that will be used by the military. A future study is to test the HCI model for GUI that will be used by people in other areas, e.g. hospital and banks to see if it works there too.

Another future study could investigate if the speed of the interaction with the system through the HCI will have any impact on the user's memory. Possibly, the user could forget if a goal has been completed or not, because the execution time was so small and therefore it can be difficult to remember.

7.2. Appendix B

The code used to get the grid coordinates of the GUI.

```
1. package model;
2.
3. import javafx.event.EventHandler;
4. import javafx.scene.Node;
5. import javafx.scene.control.TableColumn;
6. import javafx.scene.input.MouseEvent;
7.
8. /**
9.  *
10.  * @author jonas
11.  */
12. public class getMuseCoordinates {
13.     private Node node;
14.     private TableColumn colum;
15.
16.     public getMuseCoordinates(Node node){
17.         this.node = node;
18.
19.         node.setOnMousePressed(new EventHandler<MouseEvent>() {
20.             @Override
21.             public void handle(MouseEvent event) {
22.                 System.out.println("-----
23. -----");
24.                 System.out.println("Node");
25.                 System.out.println("Mouse X-cordinate: "+event.getSceneX());
26.                 System.out.println("Muse Y-cordinate: "+event.getSceneY());
27.                 System.out.println("Width: "+ node.getBoundsInLocal().getWidth());
28.                 System.out.println("-----
29. -----");
30.             }
31.         });
32.     }
33.
34. }
```

7.3. Appendix C

The script the potential user did to try the GUI.

Test of GUI with a potential user

Mode: Plan

Step 1: Rank the Missions

Task: Rank the missions in the following order:

1. Mission: Defend musköbasen, Organization: Amfibiebataljonen
2. Mission: Medivac, Organization: Skyttebataljonen
3. Mission: Standard operation, Organization: Livbataljonen
4. Mission: Defend the hill, Organization: 7 Bataljonen

Step 2:

Task: Change the following properties at these missions

1. Mission: Medivac, Organization: Skyttebataljonen
P-2-P: Between *Soldier* and *Military Hospital*
Communication type: *BFT*: priority to *high* and quality to *medium*

Mode: Live

Task: Locate mission with nodes that don't operate at the desired level.

1. Edit the properties for *standard operation* and its node *Military Hospital* communication type *Control* to *Low*
2. Mission: Defend Musköbasen, make a P-2-P connection between the *Local UAV* and *Soldier*, with priority and quality to high.

8. Reference

1. Yang, X. and G. Chen. *Human-Computer Interaction Design in Product Design*. in *2009 First International Workshop on Education Technology and Computer Science*. 2009.
2. Zhao, X., C. Zhou, and W. Huang. *Smart home power management system design based on human-computer interaction model*. in *Proceedings of 2013 3rd International Conference on Computer Science and Network Technology*. 2013.
3. Chao, G. *Human-Computer Interaction: Process and Principles of Human-Computer Interface Design*. in *2009 International Conference on Computer and Automation Engineering*. 2009.
4. ten Hagen, P.J., *Critique of the Seeheim model*, in *User Interface Management and Design*. 1991, Springer. p. 3-6.
5. Cooper, A. and R. Reimann, *About Face 2.0*. The Essentials of Interaction Design, 2003.
6. Dix, A., et al., *Human-Computer Interaction (3rd Edition)*. 2003: Prentice-Hall, Inc.
7. John, B., *Why GOMS? interactions*, 1995. 2(4): p. 80-89.
8. Sharp, H., Y. Rogers, and J. Preece, *Interaction design: beyond human-computer interaction*. 2007.
9. Rashid, U., et al. *Designing interactions using OAI model: A new interface modeling paradigm*. in *2009 International Conference on Emerging Technologies*. 2009.
10. Shneiderman, B., *Designing the user interface: strategies for effective human-computer interaction*. 2010, Pearson Education India.
11. John, B.E. and D.E. Kieras, *The GOMS family of analysis techniques: Tools for design and evaluation*. 1994, DTIC Document.
12. Kieras, D., *A Guide to GOMS Model Usability Evaluation using NGOMSL*. Retrieved April 1, 2003. from University of Michigan. Electrical Engineering and Computer Science Department FTP site: http://www.eecs.umich.edu/people/kieras/GOMS/NGOMSL_Guide.pdf, 1996.
13. Preece, J., Y. Rogers, and H. Sharp, *Interaction Design: Beyond human-computer interaction*. 2015, John Wiley & Sons.
14. MacKenzie, I.S., *Fitts' law as a research and design tool in human-computer interaction*. *Hum.-Comput. Interact.*, 1992. 7(1): p. 91-139.
15. SYMBOLOGY, J.M., *DEPARTMENT OF DEFENSE INTERFACE STANDARD (MIL-STD-2525D)*. 10 june 2014.

TRITA -STH 2017:41