Exploring Visualisation and Learning-
Prototyping for Future Air Traffic Management Solutions

Lars Ivar Kleven

Master thesis
60 credits

Department of informatics
Faculty of Mathematics and Natural Sciences

UNIVERSITY OF OSLO

August 2016

2016

Exploring Visualisation and Learning – Prototyping for Future Air Traffic Management Solutions

Lars Ivar Kleven

http://www.duo.uio.no/

Print: Reprosentralen, Universitetet i Oslo

Abstract

Current air traffic management (ATM) technologies have reached their limitations as far as efficiency goes and is due for a renewal. SINTEF in collaboration with the Single European Sky ATM Research (SESAR) project have developed an algorithm of which purpose is to identify better solutions to ATM further steps ahead than air traffic controllers (ATCOs) are able to do with today's technologies. An overhaul of European airspace creates a need to research possibilities on how the training of future ATCOs can be renewed as well. This thesis investigates the cognitive complexity required from ATM together with automation as a possible solution to decrease it and thus making it easier to learn. Overall the thesis aims at developing a tool using the advantages of visualization and animations in order to promote learning of *separation*, a concept within ATM. The tool enabled users to influence the sequence of different types of aircraft based on separation rules that was defined within the tool itself. The results from the evaluation put forward indicate that learning is achieved, yet more testing are required. The research conducted in this thesis could be useful for future studies on next generation ATM systems. Furthermore, the challenges and design process put forward here are highly transferrable to development of interactive and educational tools in other domains.

Keywords: Air traffic management, air traffic control, visualisation and learning, separation, automation

Acknowledgement

I would like to thank my supervisors Amela Karahasanovic and Patrick Schittekat at SINTEF for their guidance and encouragement throughout this thesis. I would also like to thank the participants who participated in the expert-review and the usability testing sessions, providing excellent feedback during development of the tool.

Table of contents

List of acronyms:

Acronym	Description
AGL	Above Ground Level
ATC	Air Traffic Control
ATCO	Air Traffic Controller
ATM	Air Traffic Management
EFPS	Electronic Flight Progress System
GUI	Graphical User Interface
HCI	Human Computer Interaction
IFR	Instrument Flight Rules
MSL	Mean Sea Level
NextGen	Next Generation Air Transportation System
RSMLT	Runway Separation Management Learning Tool
SESAR	Single European Sky ATM Research
SID	Standard Instrument Departure
SVL	Sheridan-Verplanck Level
TAM	Technology Acceptance Model
VFR	Visual Flight Rules

1 Introduction

The aviation industry have over the years experienced an ever increasing amount of air traffic since the first commercial flight took place in 1914 between St. Petersburg and Tampa in Florida, USA. Modern airports strive to be as efficient as possible to process the steady increase of air traffic that needs to run through them. The currently used air traffic management systems are aging and their limitations are becoming visible as airports are required to process more aircraft than ever before. The European project called Single European Sky ATM Research (SESAR) aims at overhauling the current European airspace in order to increase capacity as well as increasing the overall safety. Overhauling European air space by re-thinking air traffic management also demands re-thinking how training can be done in order to facilitate learning towards the new air traffic management (ATM) systems. The purpose of this thesis is to look into how to facilitate learning within the context of air traffic control (ATC).

1.1 Motivation

Today, airports are operating above their intended capacities in order to meet the demands of air travel. A good example is Soekarno Hatta International Airport located in Jakarta. The airport is designed in to process 22 million passengers, but served over 52 million in 2013. The year after, the Chinese airport Guangzhou Baiyun International Airport operated at levels close to 20% above intended capacity (Bouwer et al., 2015).

Several organisations within the aviation sector are reporting forecasts in which the amount of air travel will see a steady growth for the next twenty years. IATA[1], an international organisation with over 260 airlines as members projects an increase from 3.3 billion passengers in 2014 to 7 billion in 2035 (IATA, 2015). FAA[2] estimates that U.S. airlines which handled 756.3 million passengers in 2014 will reach 1.14 billion passengers in 2035 (FAA, 2015). Eurocontrol[3] on their hand have published four possible futures of air traffic growth, their most likely scenario showing a growth of 50% from 2012 to 2035. Eurocontrol also states in their forecast that airport capacity will be a limiting factor of air traffic growth in the future (Eurocontrol, 2015). As more and more airports are likely to reach their operational capacities, it will be likely that we will see a trend of airport congestion which might be leading to situations where *gridlocks* happen. While a congestion might just be slow movement of traffic, a gridlock means that certain vehicles, in this case aircrafts will block off

[1] IATA, International Air Transport Association.
[2] FAA, Federal Aviation Administration.
[3] Eurocontrol, European Organisation for the Safety of Air Navigation.

entire networks of traffic resulting in a complete stop. Bouwer, Maxwell et al (2015) have noted five trends that are occurring within the aviation industry as their leaders are becoming more aware of congestion of air traffic:

1) Passenger growth flattens and the city suffers overall as a result of reduced business and tourism growth.
2) Larger aircrafts are being used over shorter distances to handle more passengers.
3) Network connectivity decreases, meaning that airlines are more likely to increase the number of flights on popular destinations rather than offering new ones to less popular destinations.
4) The ticket prices go up as the demand is bigger than the supply.
5) The need for slot trading is becoming higher and much more profitable.

Due to these trends, traffic congestion might be a good thing for some airlines in the short term as it gives opportunities for increasing revenue. However, in the long term this could ultimately lead to a loss of market shares as connecting aircrafts will become more and more difficult (Bouwer et al., 2015). To meet the demands of future aviation we need to develop new technologies that are able to make airport more efficient.

As mentioned in the introduction, many modern airports today are still using aging technology with many separate systems for air traffic management purposes. There are currently two projects, the European SESAR[4], and the U.S. NextGen[5] which both aims at overhauling current ATM systems, proposing a change in infrastructure consisting of fully interoperable components (Derber, 2010). As a way to increase airport capacity and overall performance in the future, a part goal of the next generation ATM systems is to improve the decision making process for air traffic controllers (ATCOs).

The ATCOs are the people with the responsibility of guiding aircraft through the airspace as well as on the airport itself. As one can imagine, this is a job that comes with high levels of stress as one must be capable of rapid decision making in an environment where mistakes can prove to be fatal. To become an air traffic controller, one needs a large number of hours put into air traffic simulators. Even if the stress level is high in the simulator, it is likely to become even higher in real world situations where actions may have severe consequences. These simulators are expensive to operate and the nature of its construction severely limits the amount of students that can practice at once. With the improvements over existing ATM solutions that projects like SESAR and NextGen calls for, there are also possibilities of re-thinking the way ATCOs can be trained for the new technologies. It could therefore prove beneficial to research opportunities to create tools that facilitate learning of ATM to new ATCO recruits outside of the simulators in order to make the training process more efficient.

[4] www.sesarju.eu
[5] www.faa.gov/nextgen/

1.2 Research context

This master thesis part of an on-going research project taking place at Norway's largest independent research organisation; SINTEF. The organisation is highly involved with the aforementioned research project called SESAR. SESAR is a long term public-private partnership with the prospects of completely overhauling the European airspace and the current ATM systems. The reason behind this project is the approaching need for a more efficient system as the growing number of flights pushes the current ATM system to its limits. The SESAR project aims at designing a system which streamlines the existing processes involving administration of aircraft both in the air and on the ground. The project's aim is to enable a more efficient way of managing air traffic and by doing so the amount of delays and the time spent on the ground will be reduced. This will result in an increase of current capacity at airports which allows for them to become more efficient. There is a similar project with the name NextGen that are launched by the United States and the FAA.

1.3 Objective

The objective of this thesis is to explore the possibilities of introducing a training tool which enables future air traffic controllers to practice air traffic management outside of the simulators and without the need of instructors being present. A finished complete tool is thought to allow for self-assessment by the trainee after each session by comparing his or hers own performance against a suggested optimal solution calculated by a clever algorithm developed by SINTEF (Kjenstad et al., 2013). This thesis aims at taking a closer look at the relationship between learning and visualisation in an air traffic management scenario as a step in the development of a complete tool in future efforts in the SESAR project.

1.4 Scope

The topic of this thesis is exploring design opportunities for an optimization based training tool for the next generation ATM systems. The focus of the thesis lies on the effects of visualisation and how different visualisation techniques might affect the potential learning outcome of a tool designed for learning. As one can imagine, building and designing a tool which covers all the elements of ATM is too demanding and too large to fit within the scope of this thesis. Earlier work have placed emphasis towards reducing aircraft taxi time (Bessensen et al., 2014) and on calculating recommended off-block times Fatland, Furuberg et al. 2014). While the main topic of this thesis is learning by visualisation in a ATM scenario, the tool developed simultaniously with this thesis will be focusing on *separation rules*[6] for aircraft departing from an airport in order to complement earlier research while also acting as an area of improvement where learning can take place.

[6] The rules that dictate the travelling distance between different aircraft on ground and in air.

By limiting the scope, an experimental prototype can be developed in order to test the idea of a learning tool that does not require the presence of an instructor. The focus on separation rules was decided because the current way of handling separation rules at the runway are largely based on ATCOs judgements rather than following a set of strict rules. Developing and testing prototypes is an important part of new ATM research as these new systems will behave differently from current ATM systems. Thus, emphasis needs to be placed on designing good learning tools for the upcoming generation of ATCOs. Future research on ATM technologies such as SESAR and NextGen is important, not only for the airport and airlines themselves. It should also be of interest to the general public who travels by air as the goals include less delays and better utilization of current airport space, making the airports as well as the airspace able to tolerate the increasing number of aircraft better.

1.5 Research question

To achieve the objective of this thesis, exploring how visualisation techniques can be used for learning, a prototype will be developed for this purpose. In order to develop a suitable prototype, the literature review contains earlier research on the areas of how visualisation and learning can be used in a way that promotes learning. As air traffic control is a rather complex affaire, related work also contains previous research done on cognitive complexity and automation keeping the cognitive limitations of human operators in mind. By developing and evaluating the prototype, we hope to answer the following research question:

What are the key factors of design when developing a prototype for the purposes of education and training of air traffic management, and what are the key visualisation techniques that should be utilized in a learning tool in order to support and create a learning experience reflecting the dynamic situation found in air traffic management?

In order to answer this question, the prototype has been evaluated by both interface experts and users representing ATCOs in their earliest stages of training. These evaluations helps us identify the promising aspects of the prototype as well as revealing important design implications.

1.6 Contributions

The contributions made by this thesis are the following:

A tool developed to explore alternative ways of teaching the ATM concept of separation to future ATCO students. The tool consist of several visual elements aiding the user to follow a set of separation rules defined in the prototype while also considering overall situational awareness. The results, challenges, ideas of future work and the entirety of the development are described.

During the work on this thesis, studies were conducted to analyse whether the developed tool held any merit as an educational one. An expert-review was conducted on an earlier stage in the project to identify interface related issues before a final evaluation of the improved tool took place. The final evaluation took form as a usability test conducted with students having very little to non-existing prior knowledge about ATM to determine if a level of understanding had been achieved and thus, if learning had occurred.

Results from the final evaluation indicated that some level of understanding was achieved after interacting with the learning tool; however, more studies are necessary to ascertain the effectiveness and usefulness of learning ATM concepts by using a tool such as this. The results found in this thesis are believed to be a good foundation on which to base future research efforts on towards educating new ATCOs for renewed ATM technology in the future.

The results, the approach and methods used in development of the tool and the tool itself might be interesting for developers who intend to build learning tools applicable for other domains where visualisation techniques are feasible. The challenges identified during the process are also highly transferrable to projects aiming to design visual solutions in other areas.

1.7 Chapter overview

This chapter presents a brief overview of the structure of the thesis as well as a short description of each chapter and their contents. The thesis can roughly be divided into three main parts:

- Chapter 1-4: Introduction and theoretical background.
- Chapter 5-8: Development of prototype.
- Chapter 9-12: Evaluation and conclusion.

Chapter 2 gives an introduction to basic concepts within ATM in order to provide the reader with some insight in underlying concepts of the domain and context in which this research is related to. Chapter 3 is a description of earlier research topics deemed as relevant to this research. Chapter 4 provides a general description of research methods commonly used within Human Computer Interaction (HCI) research. The methods that are described are the ones used in this project and has been done to provide practitioners from other fields than HCI some insight to the characteristics of these methods.

Chapter 5 contains a description of the final iteration of the prototype, providing the readers with more understanding towards the contents of the later chapters. Chapter 6 describes the general design process of developing the tool. Chapter 7 describes the development of the tool by presenting different iterations from low-fidelity prototypes to high-fidelity prototypes. Chapter 8 describes some of the challenges recognized in developing the prototype.

Chapter 9 contains a description and analysis of an expert-review as the first of two prototype evaluations. Chapter 10 describes and presents the results from a usability testing session, the second evaluation of the prototype. Chapter 11 discusses the findings from the two prototype evaluations. Chapter 12 concludes the thesis, summarising the work presented in this thesis as well as exploring opportunities and possibilities for future work.

2 Basic Concepts in ATM

This chapter is a short introduction to some central concepts within the ATM intending to provide readers not familiar with the domain some basic knowledge about it. ATC is a service in which ATCOs provide guidance to aircraft at the airport and in controlled airspace by using ATM systems. The main objectives of ATC are to regulate and advance aircraft movement and at the same time maintain safety by complying with the separation rules which describe the minimum distance between moving aircraft. Local ATC is generally divided into three different categories based on their area of responsibility: ground control, air control and flight data/clearance. In addition there are controllers who work with area control which essentially means managing air traffic in areas extending the reach of local ATC. Section 2.1 illustrates some of the complexity in ATC, describing the different roles and systems that are commonly found within the control towers as well as giving a breakdown of an *electronic flight progress strip*. In section 2.2, an explanation of the aeronautic term *separation* in is given.

2.1 Air Traffic Control

Eurocontrol divides ATM into three separate activities: air traffic control, air traffic flow management and aeronautical information services (Eurocontrol). In this thesis, the area of interest is the air traffic control activity, which is briefly explained by Eurocontrol like this:

> *"It's the process by which aircraft are safely separated in the sky as they fly and at the airports where they land and take off again. Tower control at airports is a familiar concept but aircraft are also separated as they fly en route; Europe has many large Air Traffic Control Centres which guide aircraft to and from terminal areas around airports" (Eurocontrol).*

In the control towers on airports, there are normally multiple ATCOs working as a team in order to manage aircraft movement both on the airport itself as well as the surrounding airspace. Each ATCO may have one unique role or multiple roles depending on size and current air traffic in and around the airport. This means that control towers may operate with fewer ATCOs when there is little air traffic compared to peak hours. In order to manage all aircraft, both on ground and in the air, ATCOs relies on using multiple systems in order for them to perform effectively at their jobs. In an attempt to explain how comprehensive ATC is, an example from the third busiest airport in the UK, Manchester Airport, is shown below (Figure 1).

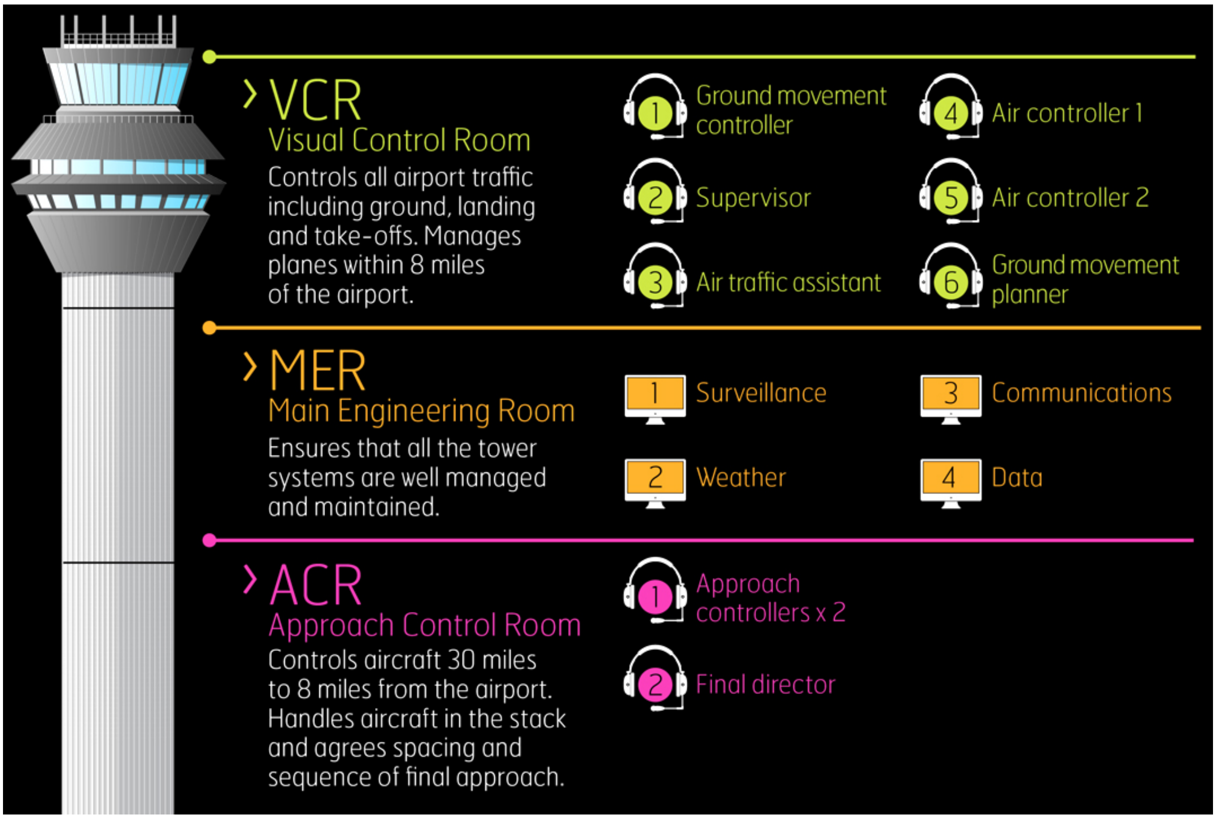


Figure 1: Description of ATCO positions and systems at the Manchester Airport, UK (NATS, 2015).

At the air control tower at Manchester Airport, there are nine ATCOs who are all working together in order to manage the high levels of air traffic in the proximity of the airport. Six of the controllers operates from the visual control room, and are responsible for managing aircraft movement on the tarmac, landings and take-offs. In this particular case, the controllers at the visual control room manage planes up to 8 miles from the airport. The remaining three controllers are located in the approach control room and their task is to manage aircraft between 8 and 30 miles from the airport which are on their way to the airport. This includes managing the so-called *holding stack* as well as maintaining spacing and sequencing for the final approach (NATS, 2015). The holding stack (Figure 2) serves the purpose of being a waiting room for planes approaching airports where a traffic congestion limits the possibilities to land. The stacks themselves are flight patterns consisting of a 180° turn, followed by a straight path before another 180° turn in the same direction as the first one, allowing planes to fly in circular patterns while waiting for clearance to land. Each stack can hold multiple planes on different levels, separated vertically by a minimum of 1000 feet. Each level in the stack may only contain a single plane. When the plane at the bottom of the stack is given clearance to initiate the final approach, the remaining planes in the stack shifts downwards. Each subsequent plane approaching the airport will then being added to the top of the stack (IVAO, 2016).

The nine ATCOs at Manchester Airport utilizes four distinct systems in order to manage the air traffic. These systems include surveillance, weather, communications and data systems (NATS, 2015). This particular setup of nine ATCOs and four systems are specific to Manchester Airport and different airports are likely to have different setups based on their local needs and practices.

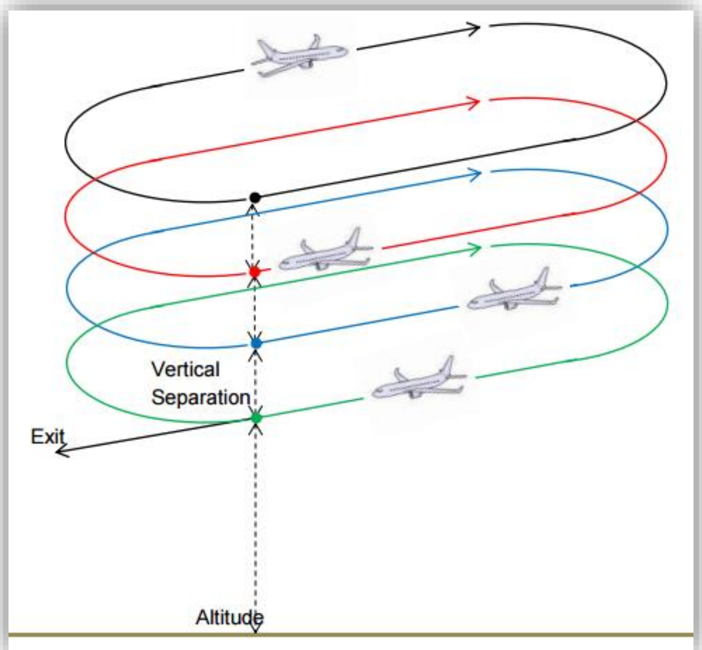


Figure 2: Holding stack containing four planes waiting for clearance to land (IVAO, 2016).

Largely, communication between the ATCOs at various airports is done verbally, and *flight progress strips* have been widely used since the adoption of commercial aircraft. A search on Google scholar revealed a patent from as early as 1950 for an invention called "Hewes Flight Progress Board" which is an attempt to improve on earlier existing flight progress boards (Hewes, 1951). The strips were traditionally made from paper and when aircrafts are moving between different areas on the airfield, it meant that one ATCO have to physically give the corresponding strip to another ATCO as the aircraft is now residing within another controller's area of responsibility.

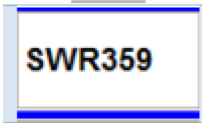
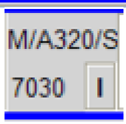
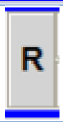
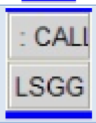
Even though air traffic across the world have increased immensely since the 1950's, paper strips are still commonly used by airports to this day. In fact, there has been raised concern about newly built airport towers not containing enough space to use the old, but trusty paper strips (Lowy, 2016). There have been made several attempts at creating better and more efficient systems by using computers, many of which are based on the already established and well known paper strip system. NAVCANstrips[7], Strip'Tic[8] and e-Strip[9] are all examples of electronic flight progress strip systems made by different companies.

[7] http://www.navcanatm.ca/en/navcansuite/navcanstrips.aspx
[8] http://striptic.fr/
[9] http://saab.com/security/air-traffic-management/air-traffic-management/e-strip/

Figure 3: Example of an electronic flight progress strip (Dolan, 2015).

This particular strip (Figure 3) is a tailored version of NAVCANstrips specifically designed for the NATS[10] tower environment and is called an Electronic Flight Progress System (EFPS). Ady Dolan, an ATCO working at Heathrow Airport in London uses this system and offered to create a detailed breakdown of an EFPS strip on NATS' own blog. In order to illustrate just how much information one of such strips contain, a breakdown of the strip, courtesy of Ady Dolan is presented in Table 1. The blue background indicates that this is an outbound strip. Arrivals use an orange background in order to easily differentiate between the two.

Table 1: Detailed breakdown of an electronic flight progress strip (Dolan, 2015).

	This is the calculated ideal take-off time and may change due to external factors such as weather or restrictions in air space capacity.
	An aircraft's call sign. SWR describes the airline (Swiss Air) and 359 identifies the specific flight.
	This is the *wake* category, which aids in identifying the correct separation distances. This specific aircraft is an Airbus A320 which is regarded as a medium aircraft. 7030 is the transponder code and the "I" means that the aircraft operates under Instrument Flight Rules (IFR).
	LOK, short for LOKKI specifies the holding point which are to be used.
	"R" for restrictions. If any restrictions were know on the aircraft's route, it would be listed here.
	Under CAL, any notes about a particular flight may be added. LSGG is an ICAO[11] location indicator describing the destination of the flight. In this case Switzerland (LS), Geneva (GG).
	27L is the runway which the aircraft will be using for taking off. 27 refers to 270° on the magnetic heading while "L" means the left hand runway.
	MID is short for Midhurst and refers to the Standard Instrument Departure (SID) route the flight will take.

[10] The main air navigation service provider in the United Kingdom.
[11] ICAO: International Civil Aviation Organization.

As can be seen in Table 1, each strip contains a wide array of different information being available in a small format that is readable for the ATCOs. Keep in mind that this is just information from one single electronic strip and that ATCOs are processing a large number of these every day.

2.2 Separation Rules

In this section, a brief explanation of separation rules will be provided as separation is one of the areas where SINTEFs ATC Optimization Service saw potential for improvement which has not been covered by earlier research on ATM regarding the SESAR project (see section 3.2.).

In general, separation rules dictate how close to each other aircraft are allowed to be at any given time during their flight. Different separation standards apply to different types of aircraft depending on whether they operate under instrument flight rules (IFR) or visual flight rules (VFR). Normally, large passenger aircraft operates under IFR while smaller aircraft and helicopters follow the VFR (Airservices Australia, 2015). It is the ATCOs responsibility to make sure that IFR aircraft are maintaining a safe distance from other IFR aircraft while aircrafts operating under VFR have their responsibility of maintaining separation placed on their pilots instead. (Schwab, 2012).

There are three categories describing different types of separation standards: vertical, lateral and longitudinal separation. Vertical separation involves a minimum vertical distance between aircraft, longitudinal separation describes minimum distance between aircraft following the same trajectory. Lateral separation is a little more complex to explain, but it is either achieved by position reports which indicates that aircraft are flying over different locations or by instructing aircraft to fly on defined tracks being separated by a set minimum angle (SKYbrary, n.d.). The reasons for separation varies from preventing collisions with other aircraft, safe distances to the ground as well as avoiding turbulent airflows which are caused by other aircraft.

3 Related work

This chapter is a literature review and begins by describing the approach taken to identify literature relevant to the thesis before presenting the relevant topics in separate subsections. In section 3.1, the process that was used to find relevant material is described together with the results. Section 3.2 contains a brief presentation of some of the findings from earlier research projects related to the SESAR project. To gain a better understanding of using visual representations for educational purposes, the relationship between visualisation learning is explored in section 3.3. Due to the dynamic nature of air traffic, animation as a specific form of visualisation and its potential to be used in the context of learning is discussed in section 3.4. The previous chapter illustrated some of the complexity in ATM, section 3.5 highlights how cognitive capabilities are one of the main factors which limits the potential for current ATM. In order to address the cognitive limitations of humans, the topic of automation is explored in section 3.6 as a higher level of automation might reduce the cognitive demands placed on human operators. Lastly, in section 3.7, the findings from this literature review is discussed.

3.1 Method

This section describes the approach taken to find literature relevant to the subject of the thesis. The search engine used was Google Scholar. Initial searches began by using keywords which could be associated with the objectives of this research and the related domain. The search returned a vast amount of hits which were covering a large amount of different subjects, many of which unrelated to the thesis (Table 2). For more accurate and relevant results, more specific searches was conducted by combining different keywords.

Table 2: Initial search terms and results (March, 2015).

Search term	# Hits
Air Traffic Control	1 960 000
Algorithm	4 010 000
Automation	2 620 000
Aviation	1 140 000
Human Computer Interaction	3 820 000
Management	5 490 000
Trust	2 780 000
Visualisation	2 940 000

Combining keywords in order to create new terms to search resulted in more accurate results and reduced the number of hits considerably. Assuming that the results provided are sorted from most to least relevant, the first three to four pages of results were considered as relevant. The relevancy of an article was judged by the title as well as the short description given on the results page. Results from this search are shown in Table 3.

Table 3: Combined keywords, results and number of relevant articles selected (March 2015).

Search term	# Hits	Selection
Air Traffic Control Visualisation	47 100	7
Algorithm Visualisation	1 750 000	2
Aviation Automation	129 000	4
Automation + Trust	170 000	9
Air Traffic Control + Management	1 430 000	5
Aviation Visualisation	33 500	2
Air Traffic Control + Trust	233 000	4
Aviation Information Visualisation	30 300	3

The gathered information were then processed, some of it kept and others discarded. Some articles were useful by themselves while other referenced to other articles which proved even more useful and relevant to the subject of the thesis. As the work on the thesis progressed and the purpose of the tool became clearer and more refined, a need to gather additional information appeared. Again, more searching was done with new sets of keywords. Results can be seen in Table 4.

Table 4: Additional search using a combination of keywords (October 2015).

Search term	# Hits	Selection
Animation + Learning	291 000	9
Animation + Understanding	338 000	3
Visualisation + Learning	103 000	4
Visualisation + Understanding	167 000	3
Cognitive Complexity + Air Traffic Control	89 800	4

The same approach was used here, the first three to four pages of results were examined closer and a selection was made based on the titles and short descriptions which are visible on Google Scholars result pages.

3.2 ATM and the SESAR project

Previous research conducted by SINTEF within the SESAR project include two projects done by bachelor students and a master thesis. The Flashback project (Bessensen et al., 2014) focused on SESARs goal of reducing time spent by aircraft taxing on the runways and to improve overall punctuality. The result of this project was a post-simulation tool which intention is to replay a simulation to the ATCO students while indicating which choices could have been made differently and why. A crucial issue uncovered in the flashback project was that the expert evaluators expressed a wish for the playback function of the simulation to actually show the preferred actions instead of just giving textual feedback. A playback showing the ATCO student the optimal solution might be beneficial towards achieving greater understanding. However, the problem is that one action made differently might influence upcoming decisions in such a way, making the playback very different from the original simulation, thus it would be harder for a student to recognize his mistakes and thus reducing the learning value of the tool (Figure 4).

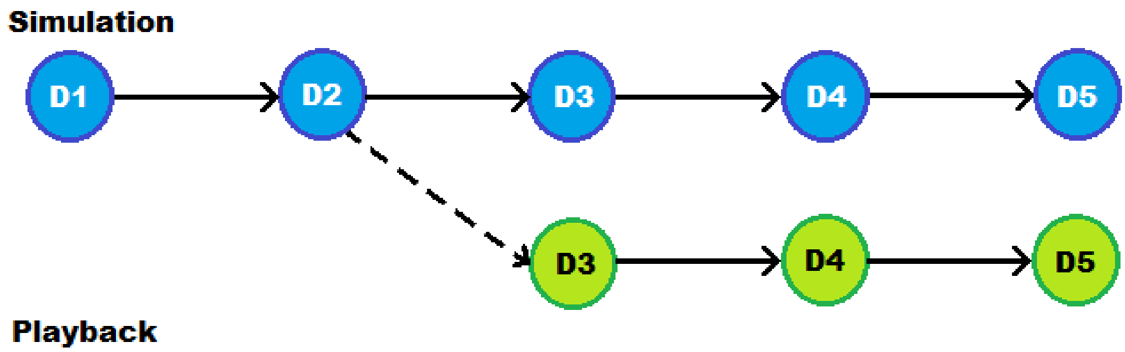


Figure 4: Illustrating a problem regarding playback of a simulation. One decision (D) might cause a ripple effect towards later decisions, making the scenario playback unrecognizable from the original simulation.

The Goodgate project (Fatland et al., 2014) was based upon creating a decision support tool that could either remove or reduce the need for an instructor in the learning process of new ATCOs. Their tool focused on calculating and notifying the user of recommended off-block time. Doing this, they found that colour was more efficient at displaying information in a clear manner to the user compared to numbers continuously counting down to zero. They used the following colour codes for their experiment:

Red	=	more than five minutes to off-block,
Orange	=	less than five minutes to off-block,
Green	=	off-block time is now.

The final solution in this project ended up using colour codes for aircraft as well as a small clock next to each aircraft showing the exact time until suggested off-block time.

Ødegård (2013) explored in his master's thesis; Exploring Visualisation Solutions for Air Traffic Control Workflow Productivity Improvement, the possibilities of using self-assessment together with an interactive visualisation tool to learn traffic patterns that emerge on airports. A tool which can be used outside control rooms was developed with the idea of generating insight in the users, supporting better decision making and creating better routines for ATCOs.

An integral part of Ødegårds work is about visualisation and insight, about becoming aware of a situation and how and where improvements can be made. Insight in visualisation can also be described as understanding the connections and representations between the elements in the visualisation, how they interact with each other and the surroundings. As insight differs on an individual and subjective level, it is hard to evaluate if insight has been generated or not. But it is suggested that users that are experienced with a certain field or domain are likely to gain a greater amount of insight than users with little to no experience Saraiya (as cited in Ødegård 2013). The thesis concludes that the tool help the users generate insight by visualisation, but it cannot be verified that the insight gained are *deep insight* because of the very limiting period of time in which this research was conducted. It is also argued that self-assessment should be limited as this can lead to false insight made by inadequate assessment by an ATCO in training (Ødegård, 2013). In the next section, we will explore insight gained from visualisation from a different perspective by the relationship between *internal* and *external* representations in visualisation and learning.

3.3 Visualisation & learning

When creating visualisation tools with the purpose to enhance learning, there are many aspects that needs thorough consideration. Diagrams, animation, colour, wire frames, pictures and graphics are all examples of visualisations, but deciding which type(s) to use are not trivial. One needs to understand when and why a certain visualisation type should be used over another and why this is better than using an alternative. To do this, much emphasis should be put on considering the user group and use context. Wire diagrams for example provide a good overview of the connections between components and how they are related, but offer no insight to the physical placement of these components. The readability and understanding of such diagrams are also likely to be different between user groups e.g. adults and children. Scaife and Rogers (1996) express that the usefulness of visualisations depends on many factors, such as experience with the representation itself, the domain of where it's applied and the type of task at hand. The terms external representation and internal representation are used to differ between visualisations that gets displayed in front of you, for example with monitors (external) and mental visualisations that are made in a person's mind (internal). To understand visualisation and learning, we first need to explore the relationship between external and internal representations (Scaife and Rogers, 1996). Meaning that effort should be placed on understanding the process that takes place when translating between visualisations and cognitive mechanisms. Ainsworth (2006) states that understanding and learning can be improved significantly if the type of representation is tailored to the demands of a given situation.

3.3.1 DeFT

Ainsworth (2006) introduced a conceptual framework for learning with multiple representations called DeFT (Design, Functions, Tasks). DeFT approaches multiple external representations concerning three aspects of learning: design parameters, functions and the cognitive operations in the learner. Ainsworth's framework consist of several guidelines for learning related to visualisations:

1. Learners should understand the form of representation.
2. Learners should understand the relation between the representation and the domain.
3. Learners may need to understand how to select an appropriate representation.
4. Learners may need to understand how to construct an appropriate representation. (Ainsworth, 2006)

The first guideline explains that one have to make sure that the learners are familiar with the form of representation. A graph or spread sheet will not be of much use if a learner does not know how to interpret them. The second guideline expresses the need for the learners to understand how the representation relates to the real world. For example, how sectors on an airport are represented visually in a tool. The third rule is specifically for multiple views, stating that learners may need to understand how to select the proper view for the task at hand. The fourth guideline concerns how learners may need to construct their own representation, as people are usually very good at their self-made representations.

3.3.2 Eight rules for designing with multiple views

In complex domains, such as air traffic control management, one can often benefit from using multiple graphical representations instead of cramming one view full of important and not so important information. Baldonado et al. (2000) introduces eight rules for designing with multiple views. The first four rules describes when to use multiple views, while the last four describes how to use multiple views.

1. The rule of diversity
2. The rule of complementarity
3. The rule of decomposition
4. The rule of parsimony
5. The rule of space/time resource optimization
6. The rule of self-evidence
7. The rule of consistency
8. The rule of attention management

Rule one, the rule of diversity applies when there is many attributes, user profiles or levels of abstraction. Rule two, the rule of complementarity is for assessing whether multiple views can help users interpret complex data. Rule three, the rule of decomposition describes how a complex view sometimes can be overwhelming and it would be beneficial to divide this complex data set into smaller isolated sets. Rule four, the rule of parsimony explains how one should limit the use of multiple views as a single view is stable while changes in views are changes in context. An example of this could be a display in an aircraft showing altitude over ground level (AGL) in one mode and altitude above the mean sea level (MSL). Additional views require a larger amount of attention by the user and therefore increase the cognitive load. The trade-off between multiple views and the higher cognitive demand needs to be weighed against one another.

Rule five, the rule of space/time resource optimization is about the spatial layout of the graphical components, whether the views are to be presented next to each other or sequentially. The rule also state that one must consider the additional resources (computing power) simultaneous views might require as well as the time might needed it takes to switch between views (loading time). Rule six, the rule of self-evidence is about making relationships between components in the multiple views visible and notable to the user. As described in the article, this can be achieved by highlighting the same object in multiple views, but the changes need to happen simultaneously on both views without delay to not cause any confusion for the user. Rule seven, the rule of consistency is about making the interfaces in multiple views appear similar to make the system easier to learn and use. Rule eight, the rule of attention management is about directing the user's attention towards the right view at the right time. To do this one must somehow guide the user towards the area of interest. This can be done by using sound cues, on screen markers, highlighting and so on, as well as combinations of these.

When it comes to designing a proposed learning tool for use in aircraft management, all of these rules apply to some extent. As stated earlier, aircraft management is a quite complex ordeal and there are many attributes that an aircraft controller needs access to, such as weight, size, times for off-block, departures, arrival, aircraft number etc. Although they do not need to be visible at all times. An additional view allowing an operator to look after all "additional" information will therefore complement the main view, which then allows for decomposition by giving the possibility of constraining the main view to show only the most used and essential information. The rules five through eight are also important when considering this is a learning tool. As sequential presentation gives some control to the designer of which aspects are shown when, an idea could be to investigate if this offer some pedagogical advantages for a learning environment compared to side by side views and if it makes sense to do so. At the same time one could investigate if rule eight, attention management can be utilized further to facilitate learning.

3.4 Animation & learning

In the previous part we looked at visualisation and learning in general. Now we will take a closer look at animation and how moving pictures impact learning in a different way than just static pictorial learning material. The dynamic attributes of air traffic suggest that animation is a well suited form of presentation to visually represent the movement of aircraft both in the air and on the ground. *Spatiovisual* is a term that is often used when discussing animation and is used to describe objects in relation to one another that can be metaphorically and/or abstractly represented by visualisation techniques. One of the key properties of animation is its effectiveness in portraying temporal movement over time (Lowe, 2003; Tversky and Morrison, 2001), something that static representations are far less suited for. Animation can be thought of as a *simulated motion picture* as done by Mayer and Moreno (2002). They define three main features of animation, including picture, motion and simulation of artificial objects whereas video on the other hand refers to movement of real world objects. Rieber (1989) suggests that efficacy of animation depends upon the learners need for one of the following attributes which animation provides: visualisation, motion and trajectory. As discussed earlier, movement and trajectory are important factors in ATM.

To explore the differences between static and animated graphics. Lowe (2001) made a list identifying possible advantages of what animated graphics can be compared to static graphics. Animated graphics compared to static can be:

- More informative because they simply show us more about the subject matter than the corresponding static depiction.
- Closer to the characteristics of actual subject matter because their direct portrayal of dynamic aspects gives a better match between the referent situation (the actual subject matter) and the representation (the way the subject matter is depicted when presented to learners).
- More explicit because every dynamic aspect is "spelled-out" directly for the viewer rather than requiring inferences to be made about the dynamics from a non-dynamic depiction. This can mean that fewer mistakes and misunderstandings occur die to there being less reliance on the learner making the correct inferences.
- More explanatory because they offer the opportunity for providing multiple visual and conceptual perspectives on the subject matter. The depiction can hence advance from description to explanation
- Clearer because unlike static graphics, they do not require various ancillary symbols (such as arrows, dotted lines, etc.) to convey the dynamic aspects of the content indirectly. This means the display can be less cluttered and learners are not required to carry out the decoding processes necessary to interpret these symbols in order to understand the changes that the subject matter undergoes.
  (Lowe, 2001)

It is argued that many animations are made without considering humans limited working memory capacity and may therefore overwhelm the user with too much information and therefore reduce the instructional effects of animation (Lowe, 2003). It is suggested that simple graphics may be more effective in certain situations, even if some details are removed from the surface (Tversky and Morrison, 2001; Rieber, 1989). Therefore it is important to design a tool which makes it easy for the user to assess the current situation in an efficient manner by looking at the main part of the screen. Still, it should be possible for the user to access further and more detailed information by digging deeper into the system if needed.

Rieber (1989) analysed the results from a study whose purpose was to compare the understanding gained from animated graphics, static graphics and text. The study did not show any differences in learning between the different groups, however it was noted that participants in the animation group used significantly less time to provide their solutions when compared to the other groups. The speed in which decisions must be made is critical in ATM and the findings made by Rieber bolster the argument for using animation in this learning tool as the most effective solution. Another study by (Mayer and Anderson, 1992) different groups (animation group, narrative group and a control group) were given instructions on how break systems and pumps function. This study concluded that animation alone did not offer any advantages, but when paired with narration, the students improved greatly in problem solving. The results of this study might be a result that have to be viewed in context of when the research was done. This article was published in 1992 and it is likely that people have a quite different approach to animation today then back then. Animation today is much more common and is found every day readily available on the internet and is something that most people are exposed to. In addition, the technology have improved immensely during the last 25 years. Computers have become much more powerful and cheaper, which allows for the production of better graphics at lower costs. In addition, there is an abundance of tools readily available that lets users create their own animations in few steps. Based on this, we expect users to be much more familiar with animation and this way of presenting information that one was in the 1990's.

Lowe (2001) states that as a result of technological improvement and the low barrier for easily creating animated content. University lecturers have taken to producing their own animations with the assumption that these will increase understanding from their students. The reason for this is that animation is a form of presentation that attracts attention as well as increasing motivation. However, there is an underlying problem that most animations are made with the focus of content rather than the attributes of animation that enhances its educational potential. Employing animation for the sake of using animation is not a guarantee for an enhanced learning experience. The educational effect of animation depends on how it is used (Mayer and Moreno, 2002). We should strive to look beyond the "eye candy" in order to create good animation with the goal of learning in mind (Lowe, 2001).

Mayer and Moreno (2002) presents a theory about using animation in multimedia presentations by providing seven principles describing how people learn. Some of which are not applicable to this particular setting regarding ATM (It is hardly doable to have a narrator describing where planes are moving on an airport). However some of the principles can be considered for this project. The spatial contiguity principle describes how students learn better if on-screen text is presented next to the object where action takes place rather than the text being placed far away. The coherence principle describes that learning is more effective when unnecessary words, sounds and animations are left out as the students' focus might drift onto irrelevant materials. The redundancy principle can basically be described as "less is more" as to much information at the same time can lead to information overload (Mayer and Moreno, 2002). Information overload might in turn lead to "load-shedding" done by the learner. Load-shedding happens when the representation (animation) requires too much processing capacity from the learner, causing the learner to limit their attention to some part(s) of the animation (Lowe, 2001).

3.5 Cognitive Complexity

In order to address the challenges in more crowded airspace, there is a need to define the meaning of airspace capacity. It is natural to think of airspace capacity as the amount of aircraft that could fit inside a given area of airspace while still withholding the defined separation rules. However, it has been identified that airspace capacity is related to ATCOs workload limitations rather than a geometric space which can encompass a set number of aircraft. (Majumdar and Polak, 2001). Majumdar & Polak (2001) defines airspace capacity using these words: *"[...] the maximum number of aircraft that are controlled in a particular ATC sector in a specified period, while still permitting an acceptable level of controller workload"*. As noted by the same authors, this definition raises the necessity to determine what controller workload really is, how to measure this workload as well as identifying the acceptable levels of workload placed on ATCOs (Majumdar and Polak, 2001). While measuring and identifying levels of workload is outside the scope of this thesis, it is important to recognize their presence and how they affect current ATC. The Merriam Webster dictionary defines workload as *the amount of work that is expected to be done.* Similar amounts of workload are likely to be perceived differently by people who are assigned said workload. This differently perceived level of workload is likely due to a number of factors such as previous experiences, age as well as cognitive differences amongst others.

As airspace capacity is limited by workload, and thus the human factor of ATC, it is important to look at ATC complexity and how it relates to controller workload. How workload is perceived is related to not only complexity of systems, but also the tasks which needs to be fulfilled in order for ATC to work. Controller workload, as the limiting factor in airspace capacity is affected by complexity in the domain of ATC and ATM systems, indicating that complexity as one of the factors currently limiting ATC. Thus, new ATM systems should emphasize on reducing complexity in order to decrease controller workload in order to become more efficient.

Complex and complicated does not necessarily mean the same thing. Cilliers (1998) distinguishes the two by saying that a system which can be described by its individual components is complicated, such as a computer. But when a system cannot be identified by looking at its components, it is complex. ATC is a complex 'system' consisting of information sourced from separate systems as well as human interaction and communication. In ATC, there are three main factors which contribute to its complexity. These factors are *airspace factors, air traffic factors* and *operational constraints* (Histon et al., 2001). While these factors represent complex properties due to the nature of ATC, they also leads to what is called *cognitive complexity*. This is complexity which affects the cognitive capabilities of humans, in this case the ATCOs.

"In order to protect controllers from situations that are too cognitively complex and, as a result, threaten the safety of the ATC system, constraints are imposed on when and where aircraft can fly. While regulating cognitive complexity, these constraints also limit the capacity and efficiency of the ATC system" (Histon and Hansman, 2008).

In research done on cognitive complexity, Hilburn (2004) identifies and presents three areas of cognitive performance which affects cognitive complexity with examples:

- Attention and memory
  Short term memory (e.g., 7 ± 2) limitations
  Selective attention limitations
  Sustained attention limitations
  Prospective memory (i.e., remembering to do something in the future) limitations
  Input and response modality compatibility
- Perception
  Time estimation errors
  Perceptual errors in closure angle
  Data code (aural vs. visual) and short term memory store compatability problems
- Decision making
  Heuristic biases in pattern recognition (e.g. shallow planning)
  Memory availability problems
  (Hilburn, 2004) .

The complex nature of ATC as well as the cognitive capabilities of ATCOs is both limiting factors in current ATM systems. While air traffic are still likely to grow in the future, the process of managing aircraft traffic will likely also grow in complexity in order to process the increasing amount of aircraft. The cognitive performance aspects identified by Hilburn (2004) above could serve as a useful reminder when designing more efficient ATM systems for the future.

3.6 Automation

Automation has become very widespread in today's modern society and exists all around us. Phones, computers, lawn mowers, heating and aircraft are just a few examples which are very likely to operate by using automated processes. For instance, the modern family car uses multiple automated and semi-automated features such as GPS navigation, headlamps that automatically adjusts beam direction and strength, automatic windshield wipers, automatic transmission, air bags, alarm system and so on. Even driverless (autonomous) cars are anticipated to become *the* way to travel in the future as large companies such as Mercedes, BMW, Tesla and even Google are promoting this technology (Greenough, 2016). A peculiar property of automation is that us as humans often do not think about or notice automatic processes at all after a period of becoming accustomed to it. It is when the automation cease to function when it grabs our attention, such as when your phone fails to automatically connect to a saved Wi-Fi network or your cars automatic windshield wipers doesn't start when it begins to rain or even worse, the windshield wipers start when it is not raining at all.

Automation is not a new phenomenon and its focus from the beginning was to increase efficiency as well as reducing labour costs as machines can work tirelessly for extended periods of time. Thomas Simpson is credited for inventing the first semi-automatic car wash as early as 1946 while the fully automated car wash got invented only five years later in 1951 by the brothers Archie, Dean and Eldon Anderson (Eisenstein, 2014). ATM as a high-risk, critical-safe domain with lots of uncertainties in its environment is not yet ready to become fully automated. Therefore, efforts towards automating processes and tasks in ATM should be focused towards assisting ATCOs by decreasing their current workload. In order for automation and humans to work together in an efficient manner, there must be some level of trust held towards the machine by the human operator. Evidently, an introduction of newly automated tasks is likely to affect an operator's situational awareness as the current work practices will be affected. The next sections will further investigate the trust-automation relationship as well as how automation may affect situational awareness.

3.6.1 Trust

The trust-relationship between a user and automation is an important aspect of automation implementation and should be explored in order to design a tool that are supposed to facilitate learning in an environment where it is crucial for operators to make good decisions in a timely manner such as in the ATM industry. Madhavan and Wiegmann (2004) explored the issue concerning trust towards automation by comparing similarities and differences between human-human versus human-automation interaction. They created two models each answering one of their two main questions: "*How comparable is human-automation trust to human-human trust?*" and "*Does the awareness of Information Source Influence trust development?*" They found that humans in general, are more adaptable in different situations while automations perform more consistently given a stable environment. They also noted that if the decision aid was human, the operator was likely more forgiving towards the error-prone human nature as well as being less observant of errors, while automated support tools are expected to perform at near perfection, thus making the operator more observant of errors (Madhavan and Wiegmann, 2004). This finding however does not take seemingly perfected automation into consideration. When a task has been automated for years and no errors of any kind have been recorded, it makes sense that an operator would become less observant as the automation never failed before and the trust towards in functioning properly is at a very high level. In their second question, the authors found no differences between human-human and human-automation interaction when the helping aid was masked. This is because the operator in this situation had no way of forming a cognitive schema, creating no expectations of the aid tool (Madhavan and Wiegmann, 2004). In their work, Madhavan and Wiegmann (2004) came to the conclusion that cognitive and psychological biases as well as response tendencies of the user are the primary constituents of trust development towards decision aid.

Trust itself is a delicate phenomenon. Trust usually builds up very slowly over time, yet at the same time it can be broken down at a much faster rate. Imagine you had a car with a function which automatically parallel parks your car. At first you would likely to be very sceptical of this function and might not even dare to try it between parked cars at first. You then decide to test this feature by letting it park your car between two traffic cones and your car parks nicely. After this you might decide to let it park between cars in the street that have a nice gap between them and gradually allows the car to automatically park between narrower gaps as your trust towards this system increases and at the same time you become less observant because the car always manages to park nicely. If the system now should fail causing your car to back into another car, your trust towards this system is likely to be greatly reduced and you might even refuse to use a system like this again.

In the article "Adaptive Automation, Trust and Self-Confidence", the authors Moray, Inagaki and Itoh defines automation as any action that could be performed by humans, but is actually performed by machine, including information processing, decision making and controlling actions (Moray et al., 2000). Endsley and Kaber proposed a more specific definition, providing four major aspects that can be automated, presented in Endsley and Jones (2012):

- Monitoring or taking in information
- Generating options or strategies for achieving goals
- Selecting or deciding what option to employ
- Implementing or carrying out actions

The amount of automation present in various environments vary greatly between different tasks. To categorize the amount of human control versus automation control, Moray, Ingaki et al. (2000) uses the Sheridan-Verplanck levels (SVL) which is a scale describing the relationships between human and automation. Lower SVL means that the human is in control while the higher levels indicate more responsibility in the automation. At SVL10, the automation makes and executes all decision without any human input, if not decided by the system (Moray et al., 2000).

Table 5: Table describing the levels of automation, based on Sheridan and Verplanck (1978)

SVL	Description of interaction
1	Human does the whole job, making all decisions. No computer assistance
2	Computer offers a full set of alternative selections
3	Computer offers a narrowed set of selections and suggests one, which the human does not need to follow.
4	Computer selects one action and human may opt to whether follow the suggestion or not
5	Computer selects action and implements it, if approved by a human operator
6	Computer selects action and informs the operator within a timeframe that allows and operator to stop it
7	Computer selects and executes an action, telling the operator what it did
8	Computer selects and executes an action, telling the operator what it did only if operator explicitly asks
9	Computer selects and executes an action, and informs an operator if it decides to do so
10	Computer decides everything autonomously, ignoring operator input (if possible)

The Sheridan-Verplanck levels can be a good indicator when trying to assess how trust is gained between systems that operate on different SVLs. On SVL 2, the computer will present a full set of choices for an operator. The operator then does not need to put a lot of trust in the computer as the operator has to select which action to take by himself. However, on the higher SVLs, the operator gains less and less control and therefore has to put much more trust in the automated system.

Even if SVLs are a good indicator of the amount of trust an operator need to place in automation, it does not, however say much about how trust is gained on the different levels. There are also other factors that impact the trust relationship between human and machines, one of them is familiarity and another is complexity. An example of a high sense of familiarity and relatively low complexity is the standard elevator. An elevator is something that have been existing for a long time, they have been tested and greatly improved since it was first invented and people are used to operating them on a regular basis. Elevators are also fairly easy to use as there are only a few buttons where most of them represent a floor and some additional buttons for opening the doors and calling for help if necessary. The task itself is also seemingly quite simple to the users as its purpose is moving a carriage up and down an elevator shaft. In general elevators can be considered trustworthy by most people and the people who frequently use elevators might even become so little observant that they even walk out on the wrong floor if the elevator stops for more passengers on its way up or down.

3.6.2 Situational Awareness

Situation awareness is an important factor when designing modern automation. A GPS tells the user something about the layout of nearby roads, providing certain situation awareness, letting the user know exactly where to make a turn. At the same time the user might become too involved looking at the device, resulting in a loss of situation awareness towards the surrounding traffic. As of today, systems and processes take use of more and more advanced automation tasks. Therefore it is important to acknowledge this complexity in automation as a major factor concerning an operator's situational awareness and whether its impact is good or bad on different levels. There is a plethora of definitions concerning the term "situation awareness", but most of them seem to carry the same essence; that it is about being aware of your environment and what it means to you now and in the future. Endsley and Jones (2012) uses the definition stated in earlier work done by Endsley and Kaber: *"[...] the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future"*. Furthermore, Endsley and Jones (2012) also break down this definition of situation awareness into three different levels where each level include the previous one(s):

- Level 1- *perception* of the elements in the environment
- Level 2- *comprehension* of the current situation
- Level 3- *projection* of future status

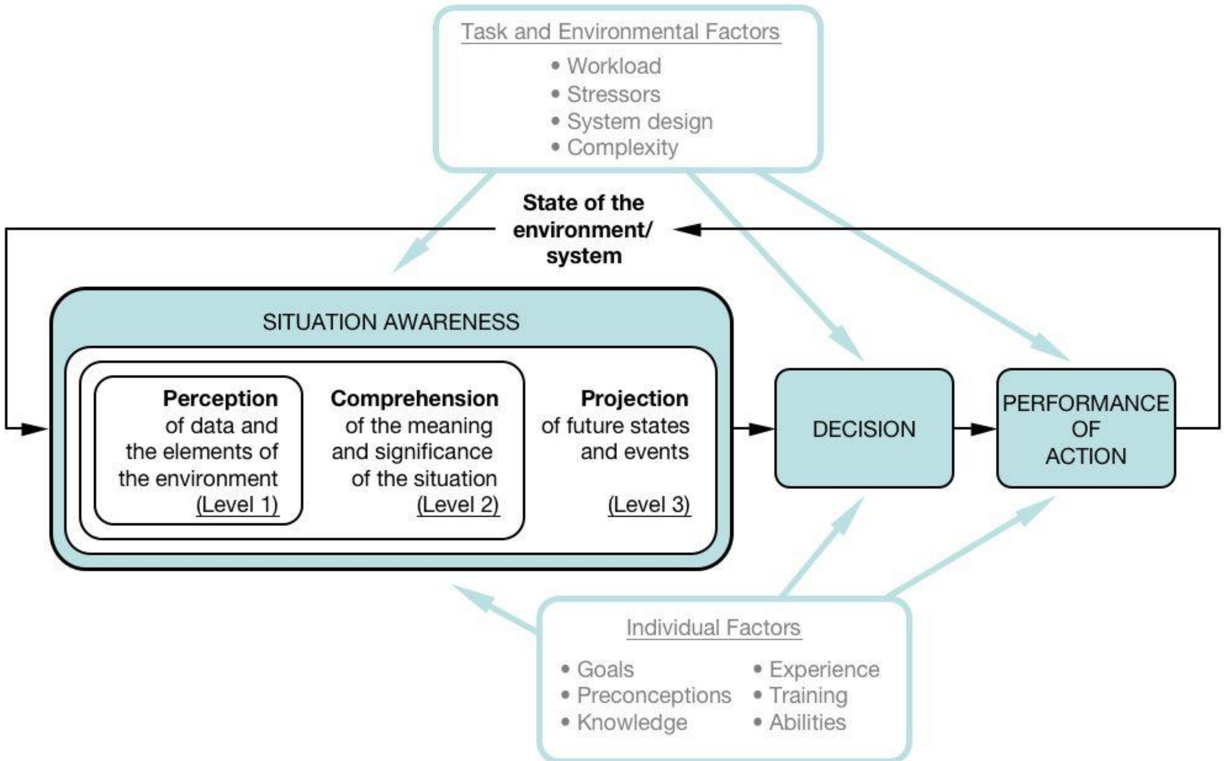


Figure 5: A synthesis of Mica Endsley's situation awareness models (Lankton, 2007)

A high level of the correct type of situation awareness is important and should ideally not reduce situation awareness in other important aspects such as in the previous example with the GPS to in to avoid unnecessary accidents. The use of automation also highlights the issue of defining who is responsible when a failure eventually occurs. Again with the GPS example; is it he devices fault for taking the attention away from the surrounding traffic, or is it the user's inability to prioritize and assess the current situation correctly that is the offender if an accident should occur? As we all know, by law, the driver would be prosecuted for dangerous driving, but the point is that when designing automation solutions, situation awareness should be taken into consideration.

As of today, systems are getting more and more complicated and operators can be limited to only work on a small part within a bigger system and therefore they might not be able to gain thorough knowledge about the system as a whole, which also might result in a worse level of situational awareness than what is desired. In addition, there are many systems today that conflict and must "fight" for an operator's attention. One example is the cockpit of an aircraft where many systems can be displayed on the same monitor, but not every systems data can be shown to the pilot(s) at once (Endsley and Jones, 2012). The pilot will then have to swap between the different systems to gain all of the information he or she needs to gain an adequate amount of situational awareness. Too many systems to monitor, creating a huge need for information processing can quickly overtake an operators capacity to process all of this information leading to a *data overload* where the human brain becomes the bottleneck (Endsley and Jones, 2012).

Another problem is that there can be very small, subtle differences between each mode, such as a single letter indicate which mode the system currently is working in. These small differences on the display can lead to operator making decisions based on the wrong information and are a likely cause of errors belonging to a lack of level 1 type situation awareness (Endsley and Jones, 2012). With the complex systems existing today that often consist of many more subsystems, there might be a need to fully automate some tasks on a SVL of 6 or higher to reduce the workload of the operator to become efficient as well as hopefully making the operator's job less stressful. One drawback about this is that automation has a tendency to reduce an operator's situational awareness leading to a reduced ability to detect faults in the automation and fully understand the state of the system (Endsley and Jones, 2012). When tasks are done automatically, an operator become less aware of those tasks and the environment an operator is working in might change unnoticed. This makes it harder for an operator to gain and adequate mental model of the situation at hand and might lead to what Endsley and Jones (2012) describes as the *out-of-the-loop-syndrome*.
The out-of-the-loop-syndrome comes with the complexity of modern automation and happens when the automation weakens an operator's situation awareness:

"When the automation is performing well, being out-of-the-loop may not be a problem, but when the automation fails or more frequently, reaches situational conditions it is not equipped to handle, the person is out-of-the-loop and often unable to detect the problem, properly interpret the information presented, and intervene in a timely manner." (Endsley and Jones, 2012)

An automation related issue is that operators frequently misinterpret what the system is doing and why it is doing it, meaning that the operators are not having a level of situation awareness higher than 1. This type of error is referred to as mode awareness or automation understanding problem (Endsley and Jones, 2012). To continue with the GPS examples, a lorry driver that navigates by using his GPS system and reaches a bridge that is too low for lorries to pass underneath. The lorry driver cannot understand why this is happening as it has never occurred before, until he later discovers that the GPS for some reason had reset to "normal mode" instead of "lorry mode", not thinking about the missing little "L" symbol that usually shows in the corner of the GPS, indicating that lorry mode is enabled. The truck driver was then having and automation understanding problem.

There is also a problem concerning decision support tools that Endsley and Jones (2012) describes as the *decision support dilemma*. This involves expert systems that are supposed to improve the human decision making, compensating for a lack of expertise and appears on SVLs between 2 and 5. However, these systems might actually work against their purpose as the time required for an operator to make a decision was greatly increased with the use of decision aids compared to how fast operators made decision without decision aids (Endsley and Jones, 2012). The correctness of the decisions does of course matter, and any system will at least some times give the wrong advice (Endsley and Jones, 2012), and when this happens, an operator can be tricked into a false sense of situation awareness if he chooses to follow this advice. If the operator notices that this is a terrible advice and acts against it, the operator's level of trust towards this machine is likely to become significantly reduced.

3.7 Summary of related work

The reviewed literature indicates that there have been made attempts at improving certain aspects of ATM by introducing various methods for visualising information in a way that is quickly recognisable and readable by an ATCO. We also discovered that different systems combating for a user's attention might lead to what is called as data overload and will be counterproductive for maintaining situational awareness. Further on, when designing systems with multiple views one must consider the trade-offs between the information made available by the multiple views and the cognitive requirements.

The gathered material illustrates how learning by visualisation techniques take place, as well as how external and internal representations relate to each other. However, the external representation should be tailored to any given situation to maximise its learning effects. In general, animation has been proved to be very well suited towards visualising dynamic situations. Likely because the animation is much closer to depicting the real world when compared to textual descriptions or static images. Further on, research shows that animation allows for a much quicker assessment of a situation than its visualisation counterparts. There have been proposed multiple guidelines for how to use visualisation and animation in a way that promotes learning. These guidelines should be considered when developing a learning tool for use within the ATM sector.

It has been established that the cognitive capabilities of humans are one of the most prominent limiting factors preventing ATM to reach its full potential. To address this issue, new solutions should place emphasis towards reducing the cognitive workload required by an operator. In order to reduce this cognitive workload, there are possibilities in designing graphical solutions which are simple enough, yet intuitively show information that the user can easily process when needed in order to address memory availability problems. Another approach towards reducing cognitive load is to automate certain functions reducing the level of attention needed to be put towards said tasks. There are some rather large challenges in automation which must be addressed. Firstly, one must identify the correct balance between automation and human interaction. Secondly, a heightened level of automated tasks may affect the overall situational awareness of an ATCO, leading to worse performance in situations where deep insight, and good judgements are required. Thirdly, as a result of the previous issue, while automation might compensate for lack of expertise, a high level of expertise might be needed in unexpected scenarios that an automation does not recognize.

The studied literature shows there is little research being done towards animation and learning specifically tailored towards an ATM scenario. This thesis aims at researching the possibilities of creating a learning environment which uses animation in order to facilitate learning without straining too much on the cognitive capabilities of the users.

4 Research methods in HCI

The first part of this chapter will describe the design life-cycle model that was used when developing the tool described in chapter **Error! Reference source not found.**, as well as the reasoning towards why this particular model was selected. For the sake of completeness, there is included a description of research methods that are well known within the Human Computer Interaction (HCI) sector for those of the readers who might not be familiar with them. By describing research methods commonly used in HCI research, I hope to offer some insight to how the literature present these methods, what their characteristics are and their advocated use.

4.1 The design lifecycle model

"Developing a product must begin with gaining some understanding of what is required of it, but where do these requirements come from? [...] Underlying good interaction design is the philosophy of user-centred design, i.e. involving users throughout development, but who are the users?" (Rogers et al., 2011).

Selecting the right development model for your project is often a non-trivial decision, especially since there are so many different ones available. In traditional software engineering, there are various models which all suit different approaches for development. Rigid models, such as the waterfall model and the v-model both encompasses a certain amount of steps to accomplish before a finished product is ready (Sommerville, 2011). There are also models that support incremental delivery such as Boehm's spiral model and the Rational Unified Process (RUP) which allows for more flexibility towards changing requirements (Sommerville, 2011). Due to globalization and rapidly changing environments, methods found in agile development are used as it enables quick software development, where working iterations of software are delivered in regular intervals (Sommerville, 2011). Common methods in agile development are Scrum, eXtreme Programming (Sommerville, 2011) and Kanban (Scotland, 2010).

A user-centred approach is often adopted in HCI research (Rogers et al., 2011). This means that the real users and their objectives should be reflected in the design of the finished product. As the models mentioned above were developed with the intentions of complying with the rules of software engineering, they are not suited for this particular project. Rogers, Sharp et al. (2011) mentions two lifecycle models that has been used within the field HCI, namely the star lifecycle model and the international standard model ISO 13407. The star lifecycle model is a user-centric model which defines evaluation as its core activity surrounded by other activities such as requirement specification and analysis (Isaias and Issa, 2015). This model suggests the notion that activities can be done at any order during a project, but each activity should be evaluated before another commences. ISO 13407 at the time of writing has been turned obsolete and is replaced by ISO 9241-210 which has clarified some issues with the previous version (www.iso.org[12]). However, Rogers, Sharp et al. (2011) suggests using a simple model (see Figure 6) to provide a framework for creativity and inspiration and is the model selected for use in this thesis.

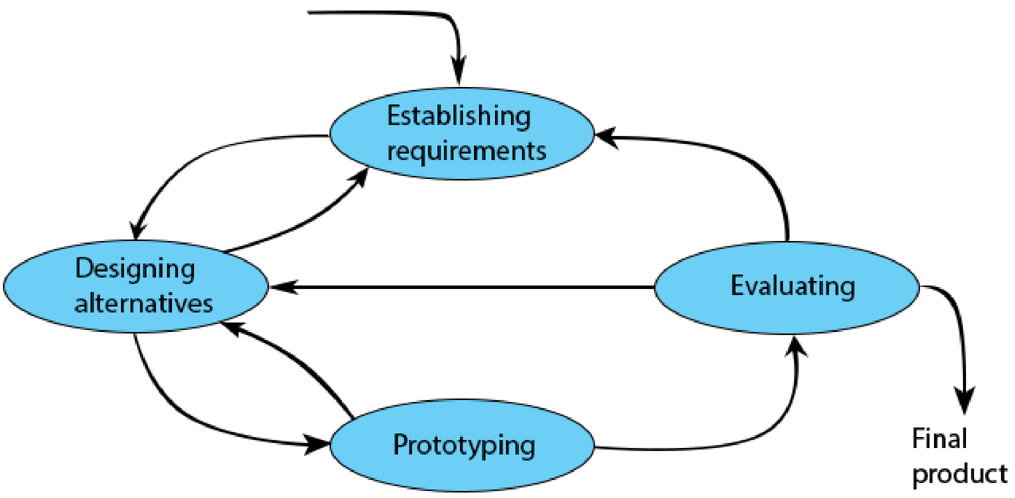


Figure 6: A simple interaction design lifecycle model (Rogers, Sharp et al, 2011 p. 332).

The simple interaction design lifecycle model pictured above allows for a lot of flexibility and includes all four basic activities of interaction design; establishing requirements, designing alternatives that matches the requirements, building prototypes reflecting the design alternatives enabling discussion and usability testing, the final basic activity is evaluating the prototypes (Rogers, Sharp et al, 2011). The model allows for an undefined amount of *cycles* before a product is finished and is only limited by time and resources. Any lifecycle model is just an abstraction of reality and should be adjusted to reflect the needs of a current project (Rogers et al., 2011). Jim Benson, the inventor of personal kanban stated during a conference that "rules kills awesome" when talking about how kanban is not intended as a specific set of rules that must be followed systematically, but should be there to give some sense of direction.

"The more rules we have, the more time we spend on checking if we are in compliance with the rules. The more rules there are, the more checking we are doing, the more checking we're doing makes us paying more attention to the rules and we're not paying attention to what we are actually doing" (Benson, 2014)

[12] https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en

Even if Benson talks about kanban in regards of system engineering, similarities can be drawn towards interaction design and HCI. Therefore selecting a fairly easy to follow and use model makes sense, especially when considering that the thesis is limited in its scope.

4.2 Approach

Collecting data is an essential part of any research and there are many existing and well defined methods for doing so. When deciding for which data gathering methods one will use, one has to consider what the data is, where and who it comes from and what is the purpose of the gathered data. The initiators of this project expressed a wish for development to be experimental and out-of-the-box thinking, meaning that the gathering of data from ATCOs and their current training environment was off-limits. This was to avoid the development from being biased on knowledge acquired about the current learning environment. In this project, the data gathering were done through the evaluation of exploratory and experimental prototypes.

4.2.1 Prototyping

Prototypes represent different states of an evolving design and are used to explore alternatives in both design and application of use (Houde and Hill, 1997). In general are two main categories to describe prototypes, *low-fidelity* and *high-fidelity* (Rogers et al., 2011). Low-fidelity is a term that describes simple prototypes that lack a lot of functionality and is likely to be very different from a final product. These prototypes can be made of anything, but are typically made of simple materials such as paper, plastic or wood. Simple paper prototype sketches can be used to explain a proposed user interface and storyboards is a good way of visualising a sequence of screens, taking a user through a task or to visualise how a certain product would be used in its intended area of application (Rogers et al., 2011). The creation of low-fidelity prototypes is especially valuable at the early stages of projects as they have several benefits for this stage in the development. They are relatively quick to build, cheap to make and as such, they can be easily tweaked based on the feedback that is produced (Rogers et al., 2011). By making a prototype too detailed in its functionality might restrict creative input from evaluators as the prototype is too well defined (Brandt, 2007), which is generally not desirable in the early stages of a project.

A high-fidelity prototype on the other hand is a well-defined, working prototype that closely resembles the final product, often built in the correct materials (Rogers et al., 2011). Thus, development of high-fidelity prototypes are bound to be more common at the mid- to later stages in product development. Even though high-fidelity prototypes are more refined than it counterparts, it has several negative aspects as it takes a long time to build and developers can get attached to their prototypes, becoming reluctant to make changes. In addition, a high-fidelity software prototype might create unrealistic expectations (Rogers et al., 2011).

Brandt (2007) states, when talking about mock-ups that the level of details are important as less details promote creativity which might lead to fundamental changes as it provoke critical feedback in both potential and drawbacks for a proposed idea. By including more details however, the focus tends to emphasise on additional features or changes within the current design. Brandt (2007) also describes prototypes and mock-ups as *things-to-think-with*, meaning that these serve as *boundary objects*. A property of boundary objects is that they limit the thought process, but at the same time emphasises focus towards this specific object and its properties. For instance, a prototype of new software is a boundary object, but even more so based on how it is presented. If this software is presented on a stationary computer, people will normally think that it is supposed to be used at one place. If the same prototype would be presented on a more mobile platform such as a tablet, people might look at it from a different angle, realising that it can be used while on the move. The form of presentation thus might be a factor towards what type of feedback a prototype might generate from its audiences.

Much of current prototyping puts strong emphasis on the properties of the prototypes themselves, how they were created with certain tools or how sophisticated they are (Houde and Hill, 1997). They propose a change in the language used to talk about prototypes to bring focus towards what role the prototype will play in a user's life, how it should look and feel as well as how it should be implemented by introducing a model of what prototypes prototype (see Figure 7). The focus should be placed on the purpose of the prototype rather than how it was created (Houde and Hill, 1997).

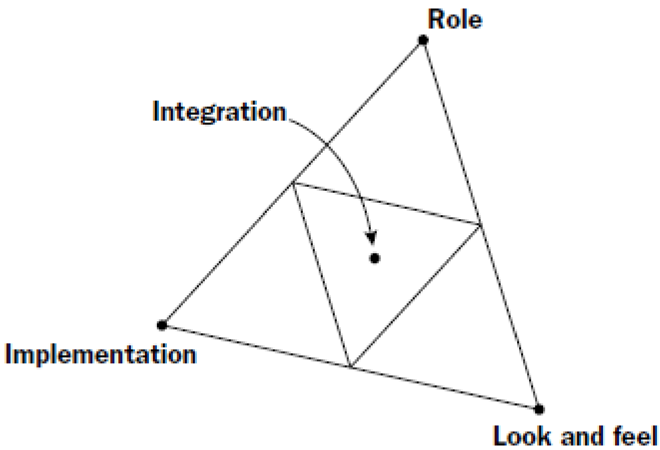


Figure 7: A model of what prototypes prototype with four categories (Houde and Hill, 1997).

This model of what prototypes prototype introduces four categories for prototyping act as a guide for what a designer should do with a prototype. In the case of the prototype made during this thesis, it should be placed somewhere on the line between *look and feel* and *role* in the above figure as the emphasis is placed towards user experience and the context in which the prototype is meant to be used rather than implementing a fully working system.

There are several advantages to building prototypes; they are useful as a way to demonstrate functionality of a concept, they allow for stakeholders to choose between several design proposals (Brandt, 2007), and they can be used as a way to clarify and validate requirements for the product (Sommerville, 2011) which might have been a misunderstanding earlier on.

Another approach taken to identify categories of prototypes include separating them into *experimental, exploratory* and, *evolutionary* prototypes (Floyd, 1984), each category containing unique attributes towards different purposes of prototyping.

The prototype developed with this thesis will be mostly exploratory as it will help clarifying requirements as well as identifying desirable and needed features. The prototype is also experimental by the definition given by Floyd as its focus also lies within ascertaining its adequacy as a learning tool for next generation ATM systems. It is however not evolutionary as it will not change too much in order to adapt to changing requirements as this is experimental research rather than building a commercial product. Combining exploratory and experimental prototyping offers a new angle to look at prototypes as they place emphasis on different aspects of prototyping and can be used to complement each other.

4.3 Evaluation methods

Evaluation is an essential part of any research that needs to be in place in order to verify the research that has been done. There are many different approaches to evaluation depending on the type of research and what the goal of the evaluation is. Rogers, Sharp et al. (2011) classifies three broad categories depending on user involvement, the level of control and the setting where evaluation takes place. These types are: (1) controlled settings involving users, (2) natural settings involving users and (3) any setting not involving users. Evaluations done in controlled settings are good at revealing usability issues, but unlike natural settings, it does not regard the actual context of use (Rogers, Sharp et al. 2011). This section will describe methods for evaluation that was deemed relevant to this thesis.

4.3.1 Usability testing

Usability testing is a good method for testing usefulness of prototypes and if they have any merit. By utilizing usability testing when developing the learning tool, one can uncover design flaws that appear as confusing for the users as well as discovering which parts of the design that is working well (Lazar et al., 2010).

Lazar, Feng et al. (2010) lists three types of usability testing: expert-based testing, automated usability testing and user-based testing. The first type means that the usability testing are being undertaken by experts in HCI following a procedure towards what is considered good design. The second type means that the usability testing is done by using some sort of software to see if certain usability heuristics are met. The third type, user-based testing includes members of the intended user group as testers for a solution. This is a good way to evaluate systems that belongs to a quite narrow and specialized area of application. As (Suchman, 2002) puts it: *"[...] in order to develop systems with any integrity, must develop them in relation to the specific setting in which they are to be used."* Considering this point of view, there are no other persons that can better represent the actual domain than the persons who resides in it. In order to do a very thorough usability testing, any and all of the types mentioned above can be combined in order to develop designs that are intuitive as well as valuable for the users. If combining types of usability testing, it is important to run the expert –based testing or automated testing before any subsequent testing with real users as it is good practice to get rid of glaring flaws in the interface before it is introduced to real users (Lazar, Feng et al. 2010). The advantage of undertaking both expert-based testing and usability testing with real users is that they complement each other. Experts are better at immediately identifying issues related to the interface itself, but they lack the detailed domain knowledge and real use application that real users have. Therefore these two methods are complementary by providing the designer with unique input from different perspectives.

Getting as many testers as possible is probably nice if your aim is to locate all potential errors that exists with a certain user-interface. However this is largely inconvenient and impractical as the benefit-cost ratio drops after reaching a certain amount of testers. A good number of users are often addressed as being five, five users will uncover about 75-80% of usability problems (Nielsen, 1995a; Lazar et al., 2010) (See Figure 8). Furthermore, Nielsen (1995) suggests that more usability testers should be used if the use-case is of a critical nature and errors may be fatal.

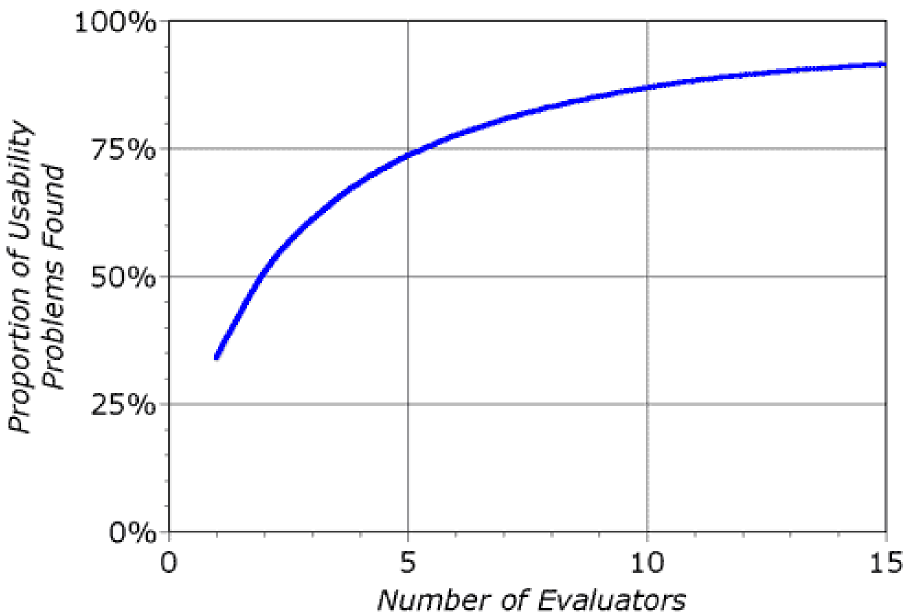


Figure 8: Curve showing the relationship between the number of evaluators and usability problems found (Nielsen, 1995a)

4.3.1.1 Expert testing

Expert based testing are basically methodical reviews performed by interface experts rather than representative users (Lazar et al., 2010). It is also suggested that it ideally should take place in the early stages of development when requirements might not yet fully defined (Möller and Tory, 2005). There are several types of expert reviews, but the most common ones are heuristic reviews, consistency inspections and cognitive walkthroughs (Lazar et al., 2010). In this section, heuristic reviews and cognitive walkthroughs will be described further as these methods will be employed in the development of the prototype.

4.3.1.2 Heuristic reviews

In a heuristic review experts evaluate an interface by comparing it against a set of heuristics that are defined beforehand (Möller and Tory, 2005). To do a heuristic review, one will often use pre-defined lists of heuristics such as the original list of heuristics as proposed by Jakob Nielsen and Rolf Molich or Schneiderman's golden rules of interface design (Lazar et al., 2010) amongst others. Usability.gov[13] have published a list of advantages and disadvantages considering the use of heuristic evaluations.

Advantages	Disadvantages
It can provide some quick and relatively inexpensive feedback to designers.	It requires knowledge and experience to apply the heuristics effectively.
You can obtain feedback early in the design process.	Trained usability experts are sometimes hard to find and can be expensive
Assigning the correct heuristic can help suggest the best corrective measures to designers.	You should use multiple experts and aggregate their results.
You can use it together with other usability testing methodologies	The evaluation may identify more minor issues and fewer major issues.
You can conduct usability testing to further examine potential issues.	

Advantages and disadvantages to using heuristic evaluations (Usability.gov[13]).

For an effective heuristic review, it is important that the experts are deeply familiarised with the heuristics themselves and they should not be familiar with the user interface (Lazar et al., 2010). As heuristic reviews are supposed to be short and relatively easy to do, it is recommended that a list of heuristics should contain between five and ten items (Rogers et al., 2011). Or "*no more than ten*" (Lazar et al., 2010). This is because having less than five is not adequately discerning, while having more than ten is making it difficult for the evaluators to remember (Rogers et al., 2011). Conducting a heuristic review does not mean that usability testing should be disregarded as findings are likely to be different. This is because while interface experts contain proper knowledge about interfaces, they usually do not possess a deep understanding of the tasks themselves to uncover issues that may arise in actual use (Möller and Tory, 2005; Lazar et al., 2010).

[13] http://www.usability.gov/how-to-and-tools/methods/heuristic-evaluation.html

4.3.1.3 Cognitive walkthrough

"The cognitive walkthrough method of usability testing combines software walkthroughs with cognitive models of learning by exploration. It is a theoretically structured evaluation process in the form of a set of questions that focus the designers' attention on individual aspects of an interface and that make explicit important design decisions made in creating the interface and the implications of these decisions for the problem-solving process." (Mowat, 2002).

In this method, the interface designer guides interface experts through a set of tasks placed within the context of a typical scenario asking several questions about usability issues in an attempt to observe if the interface makes sense. These questions will often be to observe if the correct actions are recognizable by the user, if the user-interface is clearly making visible which actions that are available and if the user's expectations matches the response given by the system while performing these actions (Rogers et al., 2011; Tsai, 1996).

Cognitive walkthroughs, like heuristic reviews can be applied in early development stages of a prototype and can even be executed with a paper simulation of the interface (Lewis et al., 1999). It should be noted that cognitive walkthroughs are suggested as a tool for developing an interface rather than validating one, and the designer should expect suggestions of improvements, further stressing that cognitive walkthroughs should take place as early as possible (Mowat, 2002). However, one should be cautious when conducting cognitive walkthroughs are being used as experts used in this walkthrough often find problems that does not appear as problems for the real users (Department of Health and Human Services Dept (U.S.), 2006). Jeffries and Miller (1991) found in their experiment about user interface testing that cognitive walkthroughs was less useful than both the heuristic evaluation and usability testing and about as effective as using a guidelines review[14]. They also noted some features that make cognitive walkthroughs unique from the other methods as it can help to define users' goals and expectations, and that the interface designers may take part in the process. The disadvantages by this method was identified as being tedious by the experts while at the same time failed to uncover some general and recurring problems, but it should be noted that the evaluators did not complete all the tasks given as this method due to time limitations (Jeffries and Miller, 1991).

4.3.1.4 Consistency inspections

Consistency inspections are conducted with one or more experts reviewing a series of screenshots from an application or web page focusing on consistency between them. Areas of interest in consistency inspections are mainly visual by focusing on layout, colours, terminology, fonts and language (Lazar, Feng et al. 2010). A graphical user interface with poor visual consistency is likely to appear as chaotic and confusing to the users, creating a negative attitude towards using the application. Keeping consistency in applications allows the users to focus on the actual task at hand instead of spending too much energy on reading and understanding the interface leading to cognitive fatigue. Thus, a consistent user interface

[14] A guidelines review is a usability inspection method that uses a vast amount of guidelines, often between 10 and 200 and takes a long time to plan and conduct (Rogers, Sharp et al. 2011).
.

promotes learning while an inconsistent interface impedes learning.

4.3.1.5 Usability testing with real users

Lazar, Feng et al. (2010) suggests three types of user-based testing; *formative testing*, *summative testing* and *validation testing*. Formative testing, generally takes place in the early stages of development and are often done on low fidelity prototypes and the focus is directed towards how interfaces are perceived rather than how a user would perform if given a task. Summative testing are often conducted on high fidelity prototypes that includes the most important functions with the aim of assessing the effectiveness of the design by giving the users tasks to accomplish. Lastly, validation testing compares the newly designed interface and compares it against set criteria found from other interfaces (Lazar, Feng et al. 2010). Usability testing with real users also allow for the evaluators to take advantage of using extra equipment such as eye tracking devices to analyse the participant eye movements searching the interface to perform his or her task. This could be very beneficial for the design of a learning tool in order to evaluate which parts of the interface is clearly understandable and visible, which elements that are hard for the user to find and also which elements that can appear as disturbances for the user. With additional equipment however, the feeling of being watched in a laboratory increases and might be a cause for changed behaviour by the participants which in turn might affect the results in contrast to field studies which is an evaluation method that takes place in natural environments with little disturbance from external sources (Rogers et al., 2011).

5 The tool (RSMLT)

This chapter presents the functionality and user interface of the tool that was developed within this project. The tool was given the name Runway Separation Management Learning Tool (RSMLT). To read about the development stages which led to this version of the tool, see chapter 7. The underlying purpose of the tool is to explore possibilities within using visualisation techniques in order to promote understanding and learning in its users. In order to achieve this purpose, the tool is designed to make it easier to comply with the separation rules which describes the minimum distances between aircraft by using visual indicators. Unlike real ATM scenarios, this tool is not divided into separate areas of responsibility such as the example from Manchester Airport in section 2.1. The tool's main goal is to experiment by using visualisation techniques in a way that supports learning while keeping cognitive demands low rather than simulating a full ATM experience.

5.1 Display breakdown

The screen consists of five main elements: main view, arrivals panel, departures panel, button panel and a panel that visualises the separation rules between the different types of aircraft. Together, the elements enable the users to perform certain tasks in a given pre-defined ATM scenario. The main goal within the tool is to issue take-off clearances with as little delay as possible. The spatial layout of the elements and the tool can be seen in Figure 9 and Figure 10.

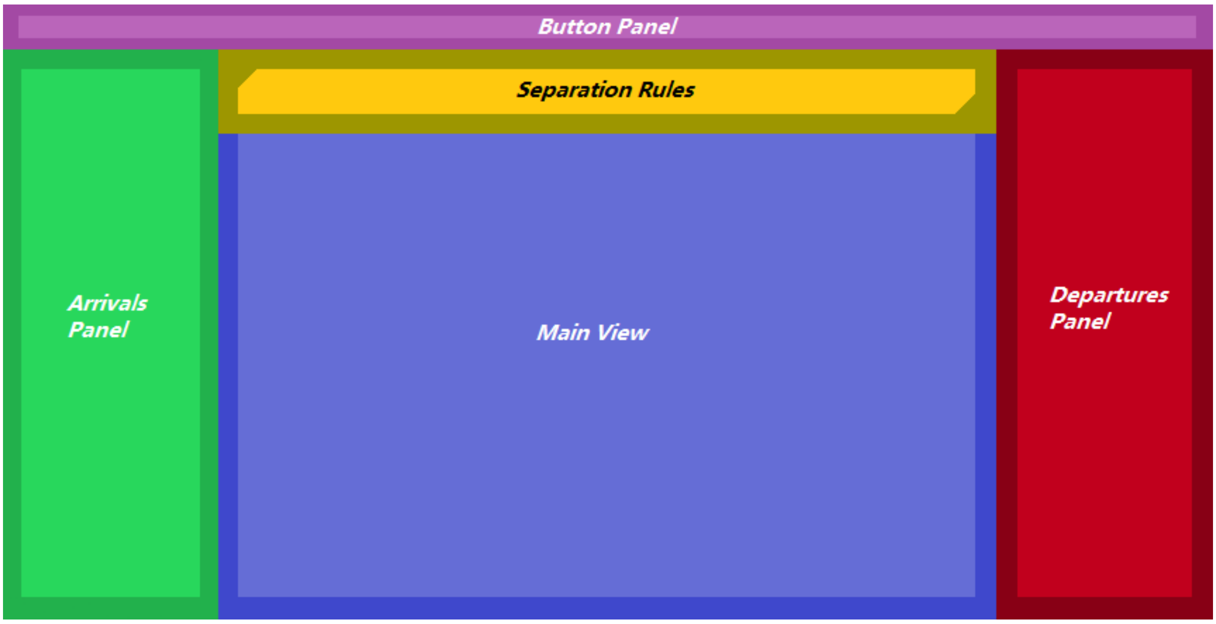


Figure 9: The layout of the five main elements of the tool.

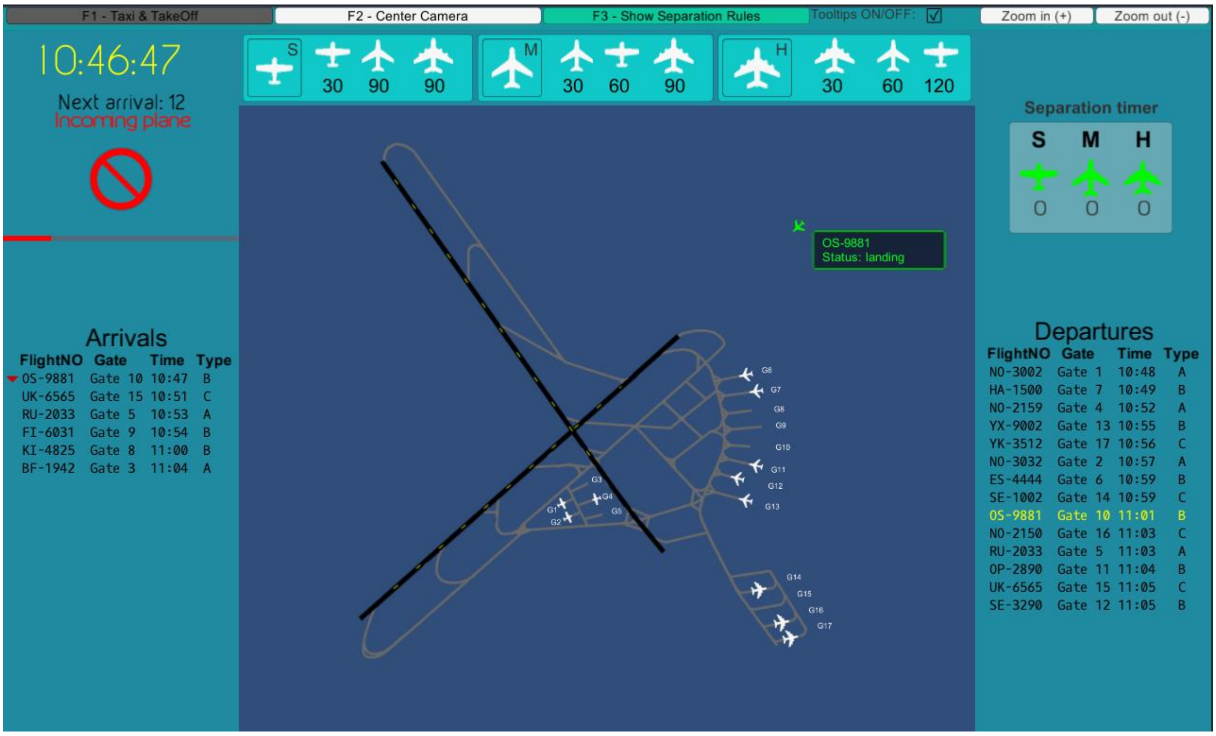


Figure 10: Screenshot from the latest iteration of the tool.

The main view is at the centre of the screen and contains an image of an airport that resembles the radar images used in current ATM systems. The airport used in the tool is based on the Hamburg Airport located in Germany, but slightly simplified. The Hamburg Airport have two runways, of which, one is used for landing aircraft and the other for aircraft that departs from the airport. The same configuration has been used in the tool with the northernmost air strip is used by departing aircraft while southern one is used for landing aircraft.

On the left side of the screen is a panel displaying information about arriving aircraft. All aircraft that are scheduled to arrive are shown in a list with basic information which includes time of landing, type of aircraft, flight number and its assigned gate number. In addition there is a visual element and a counter which tells the user exactly when an incoming aircraft will land.

The panel on the right side of the screen contains all the information about scheduled departures in a list similar to the one for arrivals. The main difference is that aircraft which are clicked on in the main view are also highlighted in the list by changing the colour of the text to yellow. This panel also contains an element called *Separation Timer*. This timer starts a countdown every time an aircraft is leaving the airport, giving the operator of the tool an indicator of how long he or she must wait until issuing another take-off while conforming to the separation rules.

On top of the screen is the button panel consisting of several buttons for operating the tool. The buttons are for issuing take-off orders, resetting the camera to its initial position, making the separation rules visible, zoom controls and a toggle switch for the enabling and disabling of tooltips.

Finally, the fifth element, the panel for separation rules is an optional panel that must be activated by pressing the corresponding button in the button panel. This element displays the rules in which aircraft must follow in order to not cause a *separation minima infringement* which describes a situation where the prescribed separation is not maintained.

5.2 Tool functions

The tool helps ATCOs in controlling local air traffic. We have here implemented only departure related functionality which are based on the idea of understanding and respecting the rules of separation. To support the user doing so, the tool has the following key functions: next arrival timer, separation rules panel and the separation timer, - all of which will be described in more detail in this seciton.

5.2.1 Next arrival timer

The next arrival timer (Figure 11) tells the user when the next incoming aircraft will land at the airport by displaying a countdown in seconds. When the counter has reached 60, meaning that the aircraft will land in approximately 60 seconds, the corresponding "lamps" will light up further alerting the user of incoming aircraft. During this time, the user cannot issue any orders for planes taking off. This is because of safety reasons due to the crossing runways. The red "reverse Ø" symbol is a well-known symbol for indicating that something is forbidden and is found almost everywhere and was selected after completing a short experiment (8.3). An example known to most people is on various traffic signs, but is also used to indicate areas where dogs, smoking, cameras, cell phones, pedestrians and so on is prohibited. The red bar underneath the "Ø" acts as an additional indicator that the runway is closed as well as act as countdown. When the indicators are lit up at 60 seconds, the bar is full and is reduced over time until it is completely faded when the timer reaches 0 and resets for the next arriving aircraft.

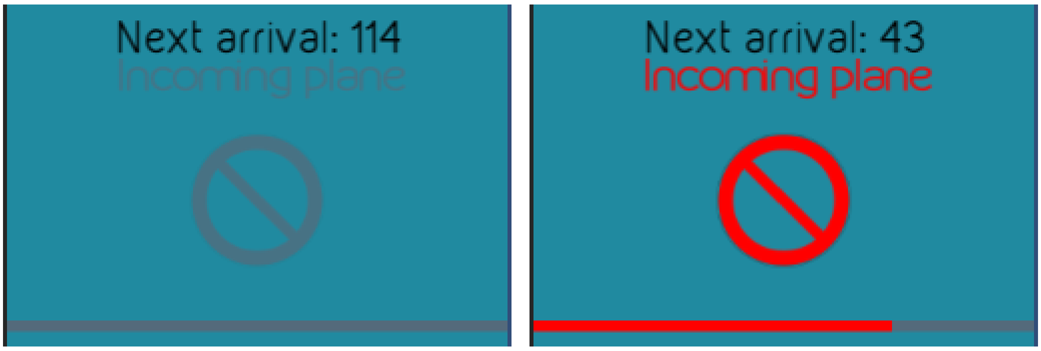


Figure 11: Next arrival timers. Once the timer reaches 60, the runways are blocked for take-off's and the indicators light up. Note that the size of the red bar is correlated to the remaining seconds in the countdown.

5.2.2 Separation rules

The separation rules are based on the general idea that minimum separation time between the different aircraft types depends on the aircraft's size. Larger and heavier aircraft is prone to produce heavily turbulent airflow behind them in the form of wake turbulence. This could be dangerous for subsequent aircraft, especially smaller aircraft with less mass. On the other hand, smaller aircraft is generally slower than larger aircraft, thus any subsequent larger planes should wait for an adequate time in order to not overtake other aircraft before they have reached their specific route. These rules are used to identify the optimal sequence of aircrafts that would minimize unnecessary waiting. The rules themselves are presented visually. Different icons correspond to different types of aircrafts (see Figure 12)

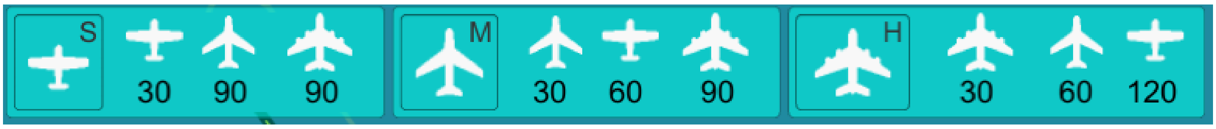


Figure 12: Separation rules for three types of aicraft.

The separation rules are divided into three turquoise boxes describing the separation rules for each type of aircraft. On the leftmost part of each box, there is an aircraft icon and a letter which represents the separation rules which applies to this specific type of aircraft. For further explanation, lets take a closer look at the medium sized aircraft. The shape of the icon and the letter M indicates that the rules for this type of aircraft is the central turquoise box. The remaining icons with the numbers underneath within the same turquoise box represents the time the user should wait before issuing a take-off orders to any subsequent aircrafts based on their type. In the case of a medium sized aircraft, a subsequent aircraft of the same type should be separated by 30 seconds, a smaller one by 60 seconds and a heavy aircraft by 90 seconds.

5.2.3 Separation timer

The separation timer (Figure 13) works in conjunction with the separation rules and functions as an indicator displaying when different types of aircraft can perform take-offs without breaking the separation rules. When take-off order is issued, the separation timer will initiate a countdown with values derived from the separation rules.

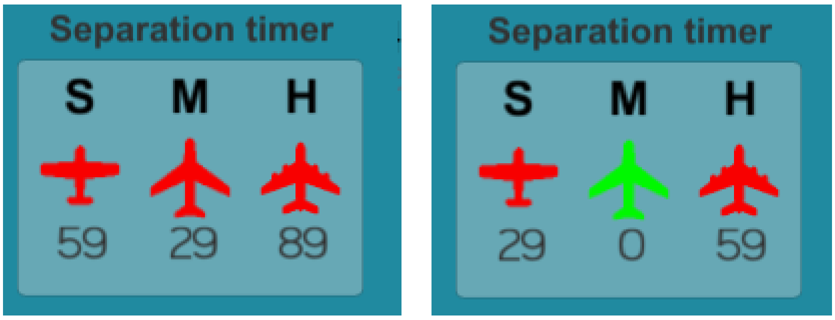


Figure 13: Colour coded separation timer.

The red colour used in the icons in the above figures indicates a no-go situation, while the green colour is signalling go. From the numbers in the left part of the figure, we can tell that a medium sized aircraft have just began departing from the airport. The right part of the figure communicates that it is clear for another medium sized aircraft to depart, but not to any of the other types.

6 Design process

The design process was begun by deciding which design lifecycle model to use in order to organize the development. It was decided to use the simple interaction design lifecycle model (Figure 6), presented in section 4.1, as this model can easily be adapted to projects of smaller scale as it is largely undefined and allows for flexibility.

First an overview of the development towards the final prototype is given by listing their descriptions in Table 6 below. The following sections each represent stages of the design process. Section 6.1 describes the process of identifying the users which is the first step in the development. Having some idea of who the target group is helps giving the development process a direction. Section 6.2 highlights the importance of identifying requirements, explains the different types of requirements and describes established requirements for developing the tool. Section 7.3 describes the activity of designing alternatives in order to meet the requirements which have been specified for a project. Section 7.4 looks into aspects of how to design solutions which aids in learnability.

Table 6: Development stages of the prototype

	Description	Section
Low fidelity prototype #1	Early sketching of what a tool might look like. Included a timeline for which users may look at the simulation similar to a video. Also included a minimap in the bottom right corner intended to highlight events of interest and their location as the main view might be focused/zoomed in on another area.	7.2
Low fidelity prototype #2	In this version the minimap was removed as it was deemed unnecessary. The graphical user interface layout was changed, due to the need for additional elements. The video player like timeline bar was removed.	7.3

	Description	Section
High fidelity prototype #1	First functional prototype was made to simulate a busy airport for 20 minutes. The menus on the left side holds information about departures and arrivals while the menus on the right holds the timers for next arriving aircraft. This version also introduced a drop down menu showing the separation rules of different aircraft types.	7.4
High fidelity prototype #2	The final iteration of the prototype in this project. Some of the issues identified during the evaluation of the previous version have been addressed in this version. The most important change in this iteration is the different grouping of interface elements with one side of the screen dedicated to approaches and the other side to departures. Tooltips were added as an optional feature to act as reminders or to offer short explanations. Additionally, markers were added to arriving and departing aircraft to give better visual feedback to the users.	**Error! Reference source not found.**

6.1 Identifying the users

As interaction design is heavily centred on users and how they should affect the design process, it makes sense to start a project by identifying the potential or intended users of a software or artefact. It makes sense to identify the users as the ones who will use the new system on a regular basis. In the case of this thesis, the users would be current ATCO students using the learning tool to gain knowledge about the new ATM systems developed as a result of the SESAR project. However, there are some complications regarding introducing replacement systems to an established user base. Introducing a new ATM system that is widely different from what current ATCOs are used to might be problematic as it is likely conflicting with current practice. *"Users don't like to deviate from their learned habits if* *operating a new device with similar properties (Norman, 1988)" (Rogers et al., 2011).* Humans have a tendency to stick with what they already know and even if they accept that there possibly are solutions that are better, they are likely to reject it as the one currently in use is 'good enough' (Rogers, Sharp et al, 2011). As it still is a long time for a tool to emerge from the SESAR project, and because of existing ATCOs possible reluctance to learning new systems, the target group for the learning tool should be new ATCO students in training with limited existing knowledge as they are less likely to have developed a specific set of routines yet.

Now that the main users are identified, who else is there? If SESAR proves successful and its implementation increases airport capacity, thus increasing the amount of passengers, there will be a lot of instances who would be affected by an overhauled European airspace. Primary, secondary and tertiary users (Eason, 1988) are categories of users used to describe their connection towards a new system. We already have our primary users identified as new ATCO students. Secondary users is described as being users who use the system occasionally, while tertiary users are those who are affected by the introduction of a system (Rogers et al., 2011). Every instance or person that have some stake in the development of a project are usually referred to as stakeholders, of which there can be many (Rogers et al., 2011).

Figure 14: Onion diagram showing different categories of users related to the SESAR project.

As suggested by Alexander and Robertson (2004) in Rogers, Sharp et al (2011) an onion diagram to model the different users involved have been used. Notice that the figure encompasses the complete new ATM system that potentially could be materialised as a result of the SESAR project, and not the learning tool which are developed in this thesis. Even if this thesis focuses on the development of a learning tool, it makes sense to look at the bigger picture as this research is being conducted as a part of the SESAR research project. From Figure 14 above, we can see that there are plenty of tertiary stakeholders who could be affected by the introduction of new ATM systems. The secondary users, by definition are users who use the system occasionally, but in the figure I have placed the airport itself and other airports with connecting flights to one where SESAR is implemented. This is because an ATM system will have no occasional users, but will only be operated by trained people who will use the systems on a daily basis. The airports themselves and their management are the ones who have to make the decision of implementing the system or not and because the airports capabilities of handling is directly affected, it is placed as a secondary user.

The above diagram (Figure 14) can be compared to the three circles of involvement model proposed by Löwgren and Stolterman (2004) (Figure 15). The model consists of three layers, core, periphery and context which are used to identify stakeholders and their effect towards the development of a project. The core represents the ones who are actively participating in the process such as the designers, developers and users who are highly involved in the development process. The periphery describes other users not actively participating in the design process, but is largely affecting it. The context describes the surrounding environment not participating in the design process, yet affecting the outcome of it. In this case the context will be official instances and laws about noise and air pollution as aircraft traffic would increase as well as some of the tertiary users identified in Figure 14 above.

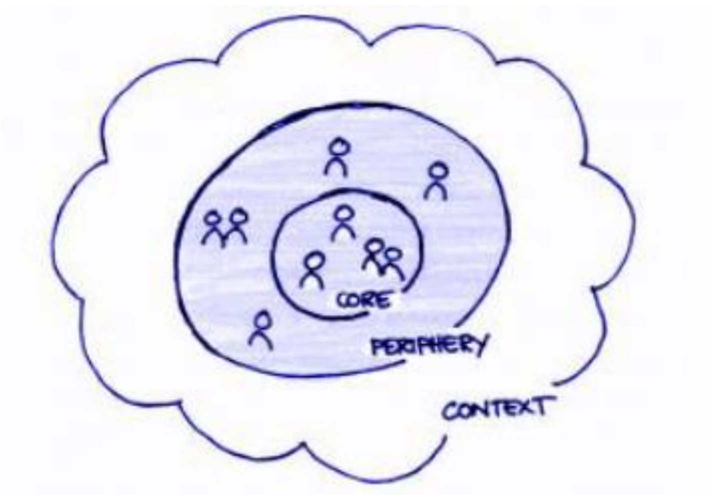


Figure 15: The three circles of involvement (Löwgren and Stolterman, 2004).

6.2 Identifying requirements

As seen in Figure 6 in section 4.1, the simple design cycle begins with identifying requirements. In general, this means that before you can design a product, one must know what is required of said product, who is it designed for and what purpose it serves. The requirements determined at the start of the project gives the product a sense of direction which the developers can stick to while building said product. In interaction design, requirements are often identified through data gathering and the analysis of these is an essential part of user-centred design (Rogers et al., 2011). There are several different types of requirements, *functional, non-functional* (Sommerville, 2011; Rogers et al., 2011), *data and environmental* (Rogers et al., 2011). Functional requirements describes what the system should do, and how it should or should not behave in certain situations (Rogers et al., 2011; Sommerville, 2011). Non-functional requirements are constraints imposed on the development of a system and not the system itself (Rogers et al., 2011). A non-functional requirement will often be time constraints due to the development of a system or legal issues regarding the system and are often likely to involve the whole system rather than explicit components of it (Sommerville, 2011). Data requirements as the name implies are requirements considering the validity of collected data. Including how relevant it is, the value of different types of data (Rogers et al., 2011). Lastly, there are environmental requirements which describes requirements established based on the context of use. Environmental requirements have four subcategories for describing which environment the requirements are involved with; physical, social, organisational and technical environments (Rogers et al., 2011).

In this project, there were three different design requirements that were established in the early stages of prototype development. The first requirement (6.3), promoting learning of a concept in ATM is a functional requirement in the sense that it describes a behaviour. The remaining requirements (6.4 and 6.5), an intuitive user interface and ease of adjusting the prototype are non-functional as they resemble quality characteristics rather than behaviours or functions.

6.2.1 Promote learning of a concept within ATM

As mentioned in section 1.4, developing a prototype to cover all aspects of ATM would be too time consuming and complex for a project of this size. This thesis, as a part of the ongoing SESAR project had the requirement to develop a tool which educates users in a concept found within the ATM. *Separation* is an ATM concept where test results from SINTEF’s algorithm for ATM optimization showed there was a big room for improvement. Separation as such are missing from previous research on ATM within the SESAR project (3.2).

6.2.2 Intuitive user interface

Before commencing the actual designing of the prototype, it was identified that an intuitive user interface is one of the key factors for making a successful product. The interface is the first thing that potential users will experience visually and therefore should look interesting, structured and inviting. The interface also plays a large part in defining how the users may interact with the software by presenting the users with visual cues about its functions and intended use. In computer technology acceptance research, the technology acceptance model (TAM) was introduced to explain determining factors of software highly influences actual use of a system (Bagozzi et al., 1989).

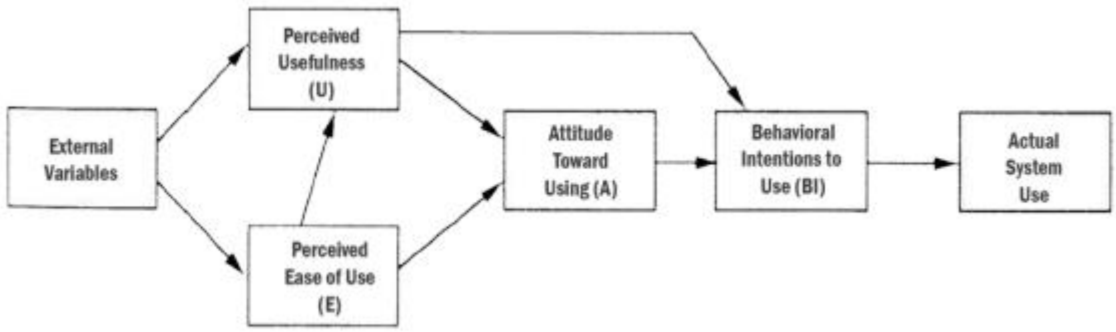


Figure 16: Technology Acceptance Model (Bagozzi, Davis et al. 1989, p.985)

TAM is based on the notion that that perceived usefulness and perceived ease of use by external variables such as demographics are fundamental aspects that directly influences actual system use through attitude towards use and behavioral intentions to use (Figure 16). Typically, computer technology are accessed through an interface which the users are required to navigate. We believe the perceived usefulness and perceived ease of use in the TAM model can be reflected by interfaces, therefore one of the design requirements were identified as creating an intuitive user interface as it correlates towards attitudes toward usage and actual system usage. Even if a system is brilliant underneath the surface, it would likely see very little use if it was not user friendly.

6.2.3 Easy to adjust

We wanted the prototype to be easily adjustable and flexible in order to make it easy to adapt to different flight patterns and also easy to change based on feedback during testing. Achieving this level of flexibility means that the prototype could be changed in order to simulate different levels of aircraft density reflecting low and peak hours of ATM. Although it is technically possible to generate air traffic scenarios when the prototype launches, we wanted to create specific scenarios in order to make the results more manageable to measure for future development. Specific scenarios would also make it easier for potential instructors to review the performance of a student.

6.3 Designing alternatives

The theory behind prototyping as a research method is largely explained in section 4.2.1. This section will mainly focus on describing the importance of designing alternative prototypes as a part of a design process.

Designing alternatives in the form of creating low-fidelity prototypes in the early stages of a project allows the designer to communicate ideas with external stakeholders through a defined and observable object rather than explaining thoughts and beliefs. Not only does prototypes help the developers and stakeholder to reach a common understanding of an idea in its current form. It also enables stakeholders to express their wishes more clearly and the developers may inform the stakeholders about complications and restrictions through an artefact.

This project is of an experimental nature with very little requirements regarding what a prototype should do, what it should look like and how it should achieve its goals. Still, designing alternatives became a very important factor during the early stages of development as the low-fidelity prototypes were tangible ideas that could be compared to one another, enabling reflection through prototyping. Through creation and reflecting on the early prototypes, some ideas got rejected while others got combined resulting in new prototypes.

6.4 Design for learnability

As the main goal for the RSMLT is to promote learning and to become a part of a learning environment, it is important to look into earlier research about designing for learnability. Jens Kaasbøll in his book Developing Digital Competence suggests a series of design actions that stimulates an increased level of learning. Some of which might be applicable to the prototype developed in conjunction with this thesis.

6.4.1 Reinforcing the user

As this tools main purpose is to increase learning and awareness in its users, it is important to focus the design on the aspect of learnability. Kaasbøll (2016) refers to behaviouristic learning theory when suggesting that learning increases when reinforcements are provided as user feedback, both positive and negative.

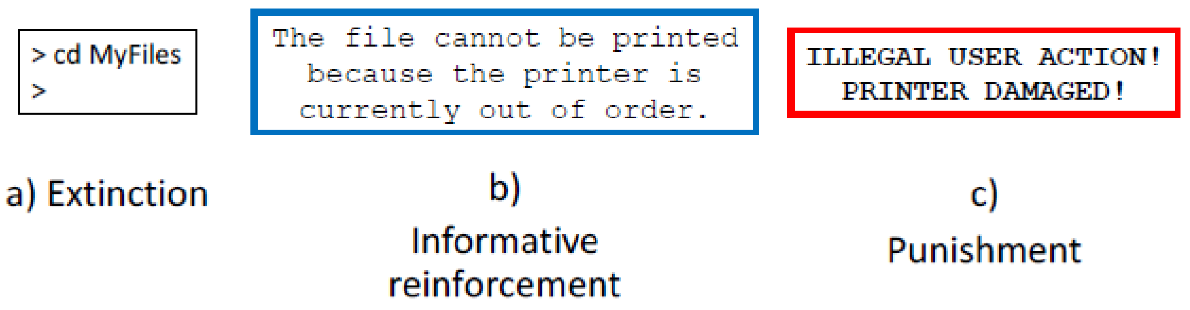


Figure 17: Three types of user feedback (Kaasbøll, 2016).

In general, informative reinforcements are better for learning than extinctions as they inform the user about the situation as well as the cause of the problem. Punishments on the other hand can decrease learning and in the example above make users wary of trying to perform the print action again (Kaasbøll, 2016).

6.4.2 Consistent design and metaphor usage

Kaasbøll (2016) suggests that learning in general will increase if new software interfaces are consistent with existing technologies or builds on previous experiences with artefacts. A designer can take advantage of using known and already established symbols connected to a specific meaning. One such example would be the much used icon depicting a magnifying glass. A Magnifying glass is often used to represent the activity of searching and is often used together with search boxes. Another example is the icon resembling a house as it is often used as a reference to going back “home” and pressing a house icon in a web browser will likely take you to your front page while pressing it on a smartphone will take you to back to the main screen. Using metaphors in interfaces, such as the house icon for “home” can be used as a bridge between existing and new knowledge and can therefore provide a new user help in understanding new software (Rogers et al., 2011). A designer which is aware of the domain in which his software will be used can exploit that knowledge by creating or use symbols that are typically known or associated with the domain itself, creating consistency with the area of application (Kaasbøll, 2016).

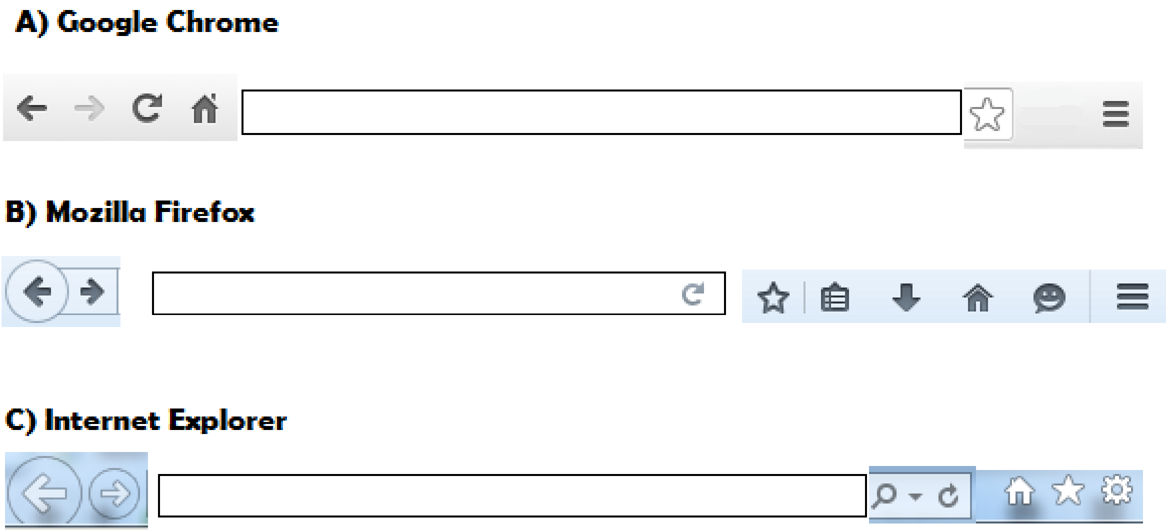


Figure 18: Symbol usage and layout of three different popular web browsers.

As illustrated in Figure 18 above, all of the browsers have a somewhat similar appearance, sharing some symbols although their placement may be different. All three browsers have arrow symbols on the left side of the search bar indicating that one can go back and forth between previously viewed web pages. All three browsers use the aforementioned house symbol as a shortcut to get to your defined start page as well as star symbols that turns yellow to indicate that a web page is bookmarked (a favourite). In addition, all browsers use the circular arrow symbol indicating to refresh a web page. Browser A and B uses the so called "hamburger menu icon" as a way to gain access to the settings, while web browser C uses a cogwheel icon. Note however that their placements are the same and that might increase understanding in a user when these two icons can be used synonymously for settings, thus promoting learning in the user.

Consistency also includes the placement of interface elements. Where does the user typically look to find certain elements? It is suggested that people do not read web-sites when looking for something, they scan them for recognizable features (Penzo, 2006). As stated earlier, people learn from past experiences, therefore, a designer should take care when deciding where to place and how to design a search box. In an article published on www.smashingmagazine.com it is suggested that search boxes should be placed on the top left or top right on every page on a website (Fekete, 2008).

This article is from 2008 and as many probably know, trends in technology are subject of changing at fairly rapid rates. We decided to conduct an experiment to see if there had been a transformation on the placement and look of search boxes. 40 random websites were visited in order to analyse the appearance of their search boxes.

Search box placement	#
Centre	4
Upper left	7
Upper centre	14
Upper right	15
Total	40

Symbol/Button usage	#
Magnifying glass	27
Button labeled "search"	10
Right arrow	1
No button	1
Crosshair	1
Total	40

Figure 19: Frequency of search box placements and submit button types.

The results from the search box experiment (Figure 19) reveals that the upper part of the screen is still the most popular location to place the search box. However, big prominent search boxes stretching across most of the width of websites have become very common with 14 of 40 web sites opting for this solution. All but one of the search boxes were accompanied with a submit button and using the magnifying glass as a metaphor for searching is the most used solution by far. Some of the websites visited in this small test also displayed the search boxes in different areas based on you being on the front page or on an article page. There were put no consideration towards what type of websites and the placement of their search boxes (i.e. forums typically having the search box upper right and online shops having them somewhere else). For the complete list of websites visited in this test, see appendix B.1 and B.2.

6.4.3 Tooltips

Tooltips are a good way to provide brief instructions on how certain features and functions work without distracting the user too much (Kaasbøll, 2016). The way tooltips traditionally work is that they appear when the user hovers the cursor over an interface object, providing the user with helpful information about said object. Generally, a tooltip is a faster way to find desired information when compared to searching for help through a help menu or a user manual. As have been mentioned before, the speed of which information can be retrieved is believed to be a key design factor of an ATM tool. Due to the dynamic nature of ATC, a user will not have the time to look through user manuals for information.

7 Prototype Development

One of the requirements for this prototype were that it should be an experimental approach to producing a learning tool for the ATM sector. This meant that I was not supposed to look to deep into existing solutions for ATM, avoiding existing knowledge about these systems to interfere with my proposed prototype. As (Bratteteig, 2004) puts it:

"[...] our understanding of the area we are designing for deeply influences what we are able to suggest as problems and solutions – our understanding of the possibilities that technology can offer steers our attention and focus and makes our understanding of the world very biased."

This suggests that limiting designers' knowledge about existing solutions should enable out-of-the-box thinking. Leading to original ideas that are not influenced by too much existing knowledge of the current and conventional systems. The development platform that was selected as the development environment for the prototype was Unity as this is a well-documented tool, supporting a large community of developers.

7.1 Development tool (Unity)

Unity is a game engine developed by Unity Technologies[15] with the intention of making cross-platform game development easier than ever before. At the time of writing Unity supports developing builds for 23 platforms, including consoles, desktop computers, smartphones, web and virtual reality goggles (Unity, 2015a).

Unity was selected as a development platform for the prototype due to its great potential and flexibility as it supports both 3D and 2D projects. It is also a widely available tool as anyone who wants to try it can download the personal edition and use it for free. As of now, according to their own website, Unity has become a very popular game development platform since its inception in 2006. Currently possessing a 45% market share of the global game engine market (Unity, 2015b) and has become the standard software development kit for the Nintendo Wii U (Helgason, 2012).

Unity also comes with an asset store comparable to App Store and Google Play used by Apple and Android phones respectively. In a similar fashion, the asset store allows developers to share their creations with other developers for a set fee or for free if desired to do so. For developers using Unity, this means that the asset store can be used as a "shortcut" in development as one can download already existing 3D models, animations, scripts, sounds etc. to implement into your own projects. To quickly describe the massive amount of content

[15] Unity Technologies website: https://unity3d.com/company

available from other sources on the internet, a search on the key words “unity tutorial” provided over 16 million hits in Google and over 1.2 million hits on YouTube. In this project, Unity version 5.1.2 was used.

7.2 Low fidelity prototype #1

The low fidelity prototypes were drawn on paper to support idea generation and to conceptualise the ideas. Figure 19 below is an early conceptualisation of what evolved into becoming the RSMLT described in chapter 5. At this stage, the prototype was indeed still in the fuzzy beginning of development and it was still unclear on what to do with the actual prototype. It was imagined that the tool could take advantage of some sort of timeline that would allow the students to go back and forth in the simulation to look things over. It was also imagined that the interface should have the menus and tables at the bottom of the screen as well as a minimap in the bottom right corner with the main view on the remaining part of the screen. The minimap was supposed to highlight important events in case they were out of sight in the main view. It was intended that the minimap also would support quick navigation to these important events by clicking on it with the mouse button.

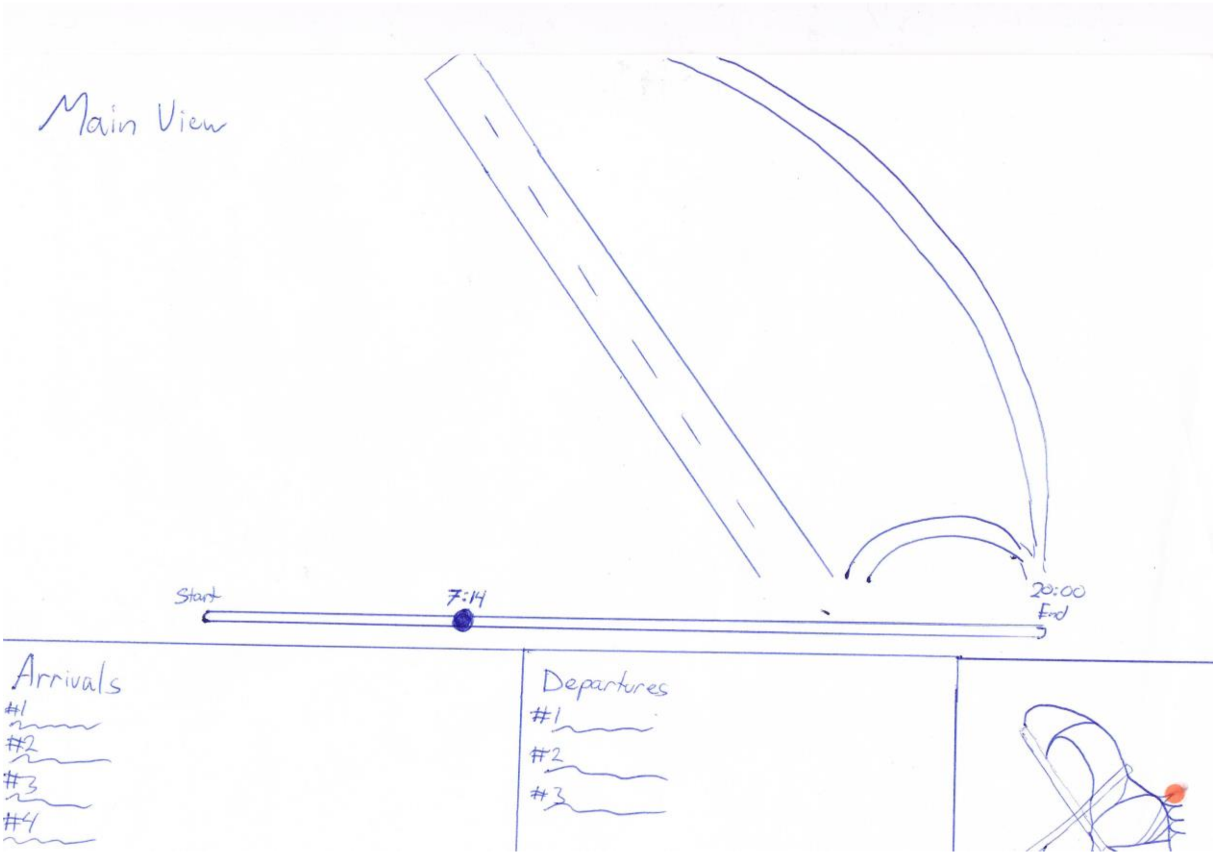


Figure 20: Early low fidelity prototype #1

7.3 Low fidelity prototype #2

In another, later low fidelity prototype (Figure 21), closer to resembling the high fidelity one that can be seen in Appendix D. The minimap was removed as it was realised that the user of the tool could pretty much have the whole airport in the main view at all times without any issues regarding visibility. The menus were moved to the top of the screen as this is a more traditional place to put buttons, which also conforms to consistency in design (section 6.4.2). The tables for plane arrivals and departures were moved to the left of the screen, while the right column was supposed to show information about a selected aircraft. This prototype also had a clock and buttons for zooming in the upper right corner.

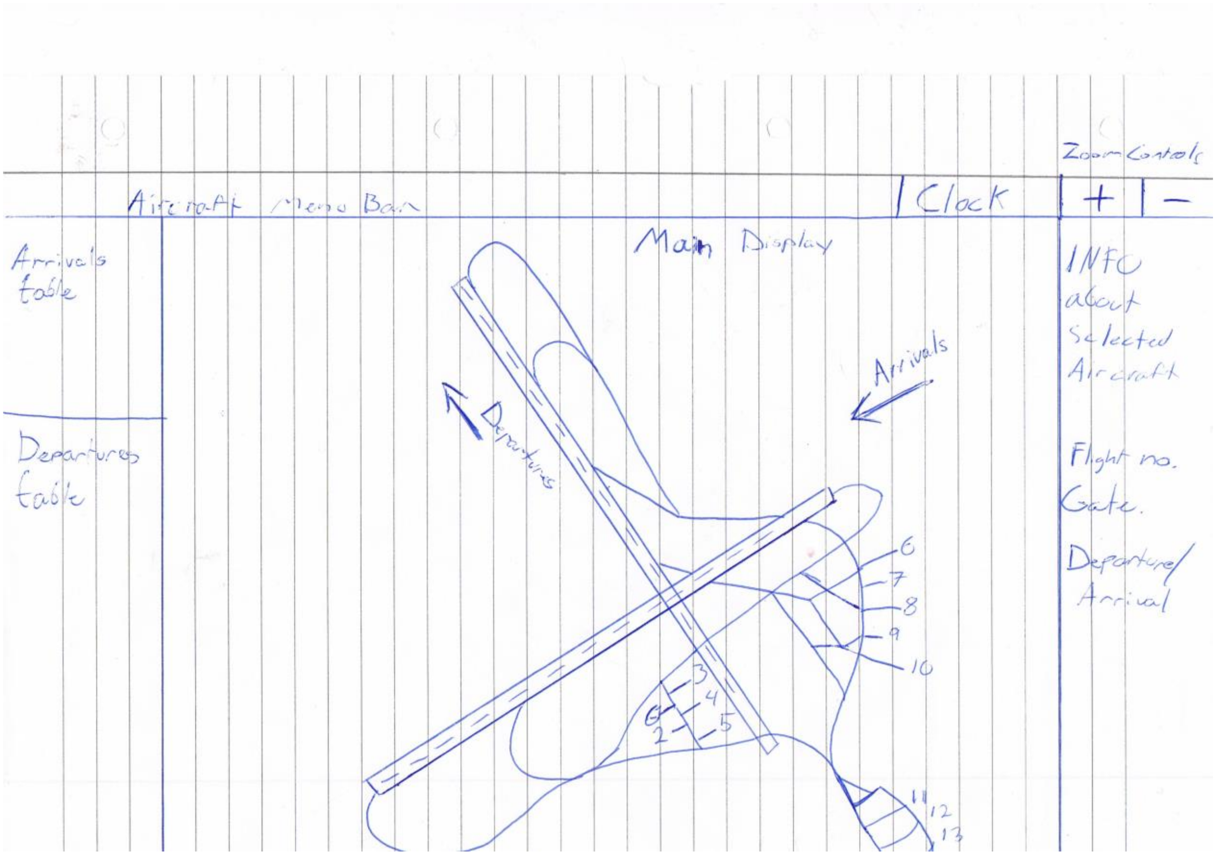


Figure 21: Early low fidelity prototype #2

7.4 Development of the high fidelity prototype

The first step in creating the prototype was to create the airport itself. This was done by using Adobe Photoshop and saving the image as a *portable network graphic* (.png) file as it supports transparency. By making the image transparent, the background colour could be easily changed for development and testing purposes. The next step was to create sprites depicting aircraft of different sizes/types, also done in Adobe Photoshop. This resulted in a so called sprite sheet (Appendix 3) containing three different aircraft designs in eight different colours and brightness each. This sprite sheet was then cut into separate sprites by slicing them in Unity's *Sprite Editor*. At a later stage, when becoming more familiar with Unity, I realised that the colour of a sprite could be changed within the Unity platform and also during runtime by scripting. Thus only three white sprites of the different aircrafts were actually used in the prototype.

Rather than writing a script from scratch enabling the aircraft to move around on the airport correctly between the runways and the gates. A script pack containing the means of creating waypoints was downloaded from the aforementioned unity asset store and slightly adapted to fit this project. The script-asset that was used was Smart 2D Waypoints[16] by Fuzzy Games Lab. Using waypoints meant that paths had to be made manually both from and to every gate on the airport. Each path is unique and consist of between 10-14 waypoints.

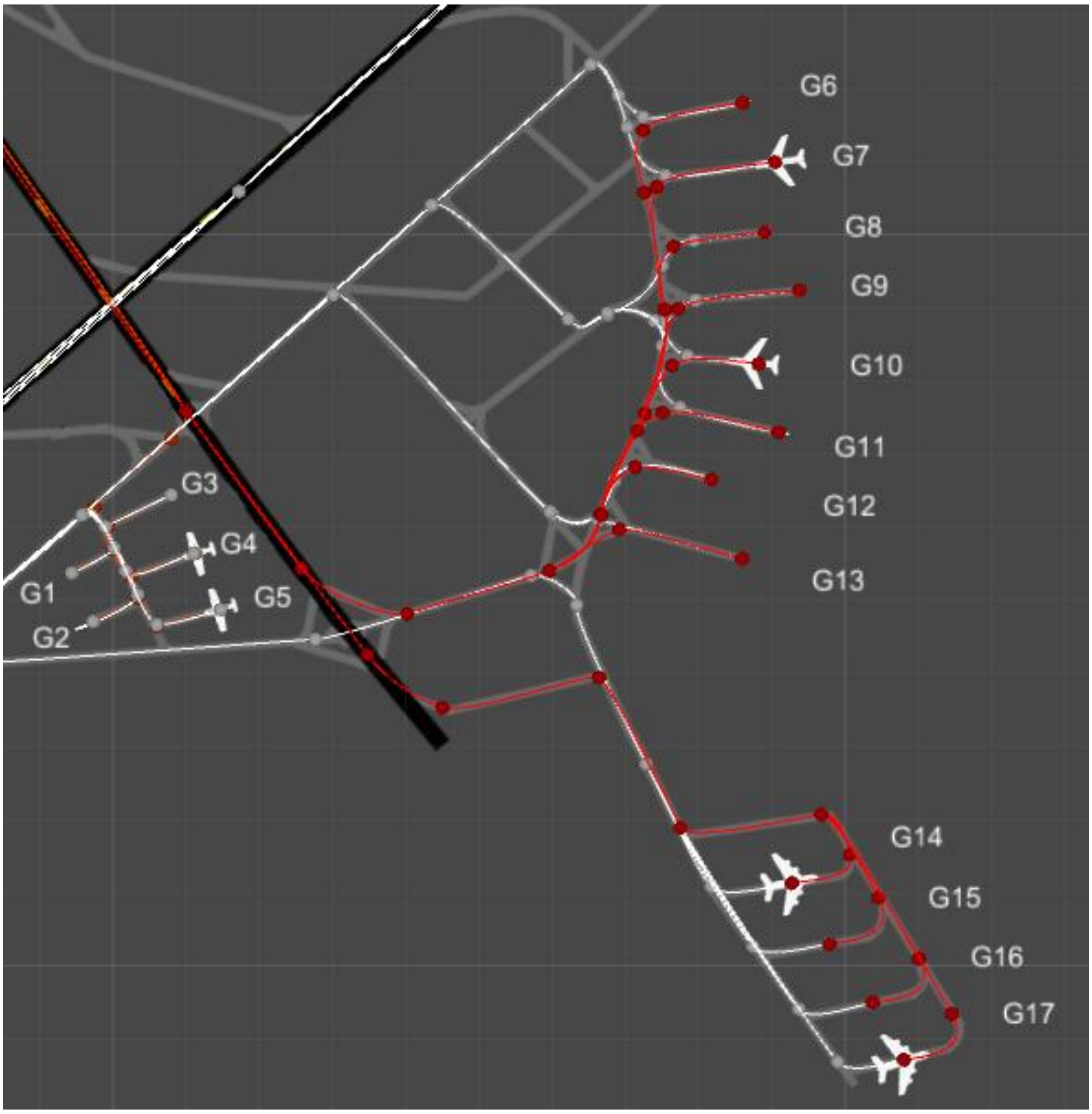


Figure 22: Bubbles are waypoints and the lines between them mark the path. White indicates paths leading to a gate while red indicates paths from the gates to the runway.

[16] Smart 2D Waypoints in the asset store: https://www.assetstore.unity3d.com/en/#!/content/15371

7.4.1 Graphical User Interface

The aim was to design a simple user interface which was intuitive and easy for users to understand. This was done by adhering to the applicable rules from the eight rules of designing with multiple views (section 3.3.2) as described by Baldonado, Kuchinsky et al. (2000), the design guidelines provided from the DeFT framework (section 3.3.1). While also keeping cognitive capabilities of humans, such as working memory in mind (section 3.5), one of the main goals was simplicity. ATC was established as a rather complex undertaking in section 3.5. If one can create a user interface which presents this complex situation as something simple, less cognitive effort is required from the user. In order to create a graphical user interface (GUI) that is as user friendly as possible, consideration was placed towards familiarity in the form of consistency and metaphor usage as discussed in section 6.4.2. When deciding where to place the toolbar and the buttons, we asked ourselves where we would expect to fund the toolbar, and this it was placed on top of the screen as is the norm in most applications.

The other elements in the GUI were grouped based on the information they contained and the type of representation. The right part of the screen contained the more dynamic tools such as the sequence timer and the next arriving aircraft indicator. Meanwhile, the left part of the screen contained the information about arriving and departing aircraft in a list format including call sign, gate number associated with the aircraft, time of arrival, time of departure and type of aircraft. Aircraft types were differentiated by different icons together with an assigned letter letter:

- A = Small aircraft
- B = Medium aircraft
- C = Large aircraft

This was done in an attempt to follow rule #6 from Baldonado, Kuchinsky et.al (2000)'s set of rules: the rule of self-evidence. The letters describing aircraft types and their respective icons were used to represent the same object in order to strengthen the bonds between them in order to make their distinction more clear.

The last part of the GUI is an optional tab which acts as a reminder of the separation rules so the user can reference them if needed. The panel showing these rules is located at the top of the screen underneath the buttons bar and can be toggled on and off with an on-screen button or by the corresponding function key.

7.4.2 Scripting

To get the prototyping to actually work and have some logic implemented, scripting needed to be done. Unity supports three scripting languages, all of which can be used interchangeably in a Unity project. JavaScript, C# and Boo. Naturally only one language can be used within a single file as the file extension and thus file types are different between them (.js for JavaScript, .cs for C# and .boo for Boo). For consistency and to avoid confusion in the file hierarchy as well as confusion in writing scripts, only C# was used in this project. Some exceptions may occur when downloading assets from the Unity asset store as the provided script is usually only provided in either scripting language. Following standard coding conventions, the code was written in separate files based on their purpose and function within the prototype. An organized hierarchy of files makes it more convenient when changing existing scripts, adding new features as well as making it easier to fix bugs that appear due to faulty scripting.

7.4.3 Problems during development

There were several problems during the creation of this prototype. I will present a few noteworthy of them in this section.

Problem #1:
Some of the tutorials and information available are made with earlier versions of Unity without explicitly stating which version was used. This caused a problem when referencing material from older versions of Unity as some features behaviour, location were changed while others were no longer supported and removed entirely.

Problem #2:
Using the same waypoints on multiple paths caused tangles, knots and wonky paths as all the paths that originated from one waypoint could not be placed on top of each other. The solution was to create separate waypoints for each and every path, placing the waypoints directly on top of one another.

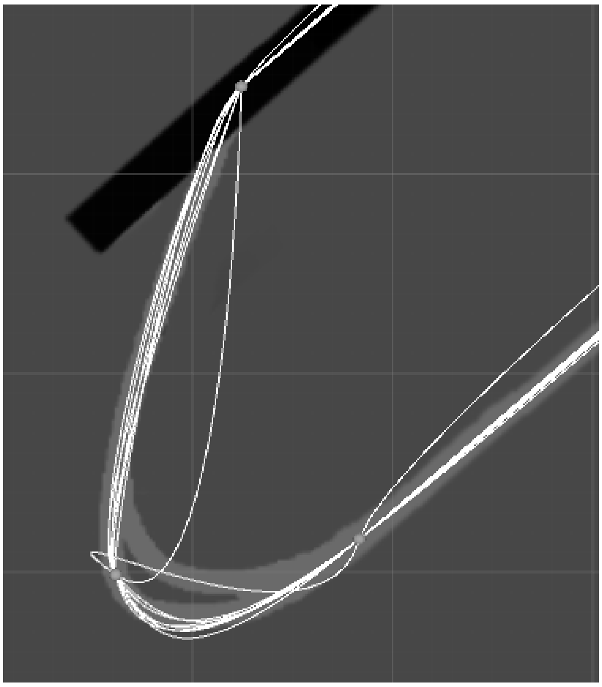


Figure 23: Paths behaving weird caused by too many paths going through the same waypoint.

Problem #3:
Initially different screen resolutions were not considered at all causing the interface of the tool to look and behave weird on resolutions differentiating from the one used when developing. The result was a bad looking GUI with overlapping elements making it hard to read and separate them. Lower resolutions also made other elements look cluttered. This was solved by making GUI elements anchored within the GUI canvas and their size are set as a percentage of the screen size.

7.5 Development of high fidelity prototype #2

The evaluation of the high fidelity prototype in the previous section uncovered multiple issues with the tool. These issues were analysed and resulted in some changes and improvements over the previous version. The changes made from the earlier version of the prototype will be presented in this section. The results from the evaluation that influenced the changes presented here, are described in chapter 8.

The aircrafts type-letters were changed in order to better reflect real classification of aircraft sizes. It was made clear in the evaluation that the proper classification is small, medium, heavy and super-heavy for describing aircraft. As this tool only encompasses three classes of aircrafts, the new aircraft letters are the following:

- S = Small aircraft
- M = Medium aircraft
- H = Heavy aircraft

One of the main issues was the difficulty of understanding the separation rules. It was stated that it was difficult to read them as they were sorted on aircraft type. Instead, the separation rules are now sorted by time-value for separation rather than aircraft type. In addition the type-letters have been included in a further attempt at highlighting the connection between interface elements, following Baldonado et al's rule number 6, the rule of parsimony (section 3.3.2).

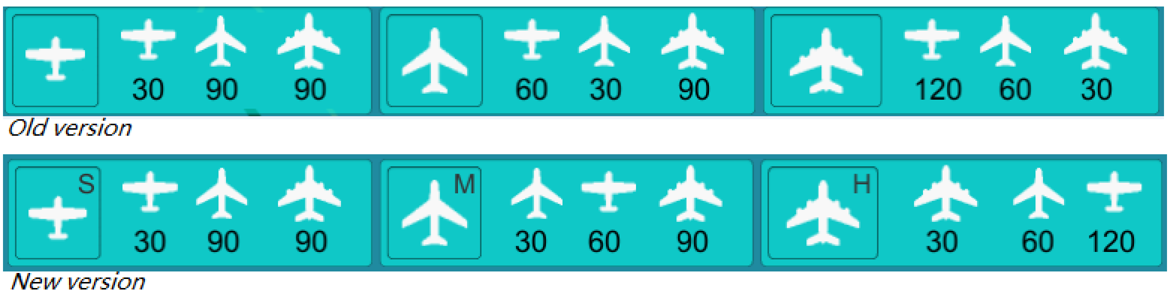


Figure 24: The differences between the new and old presentation of the separation rules.

To further highlight when aircraft can perform take-off's while still adhering to the separation rules, colour coding was introduced to the separation timer. Red means that this type of aircraft should ideally wait for a set amount of seconds before taking off. The wait time in seconds are displayed underneath the icons for each aircraft type. A more detailed description was given in section 5.2, describing key functions of the tool together with Figure 13 showing what it looks like.

Another problem the experts encountered during the expert-testing was that it was hard of them to understand where to look in order to locate the information they were asked to find. To address this issue, the left and right information panels were re-designed. The previous version held lists over arrivals and departures in one panel and the "tools" in the other. This was changed to the leftmost panel holding all the information about incoming aircraft while the rightmost panel holds all the information about departures. The intention for the restructuring of interface was to make it easier for users to find the information they are looking for more efficiently. Some of the experts thought it confusing that aircraft disappeared from their respective lists the moment they either landed or started to depart from the gate. To give better inform to the user of what is happening, when either of the aforementioned situations happen, a little arrow was placed next to the aircraft in the list, disappearing after 30 seconds. The arrow is there to give the user visual feedback that a plane is landing, has just landed or is leaving the airport.

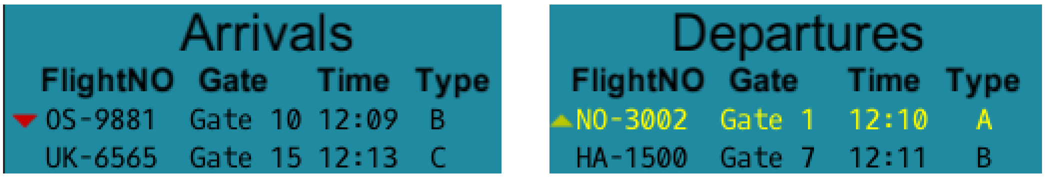


Figure 25: Arrows indicating status of aircraft.

An aircraft coming in for landing is indicated by a downward red arrow, when the aircraft has landed the arrow turns yellow for 30 seconds until the list item disappears. A yellow arrow pointing upward indicates that this aircraft is about to leave the airport and will disappear from the list in 30 seconds.

To adress the understandability some of the experts were concerned about, tooltips were added to certain elements, providing short explanations to aid the user. As described 6.4.3, tooltips is an easy and quick way of accessing help compared to other methods. In order to make the tooltips appear, the mouse pointer needs to be hovered over the interface element for a short while. Experienced users and those who does not want to use them may disable them by toggling a checkbox.

Notifications (Figure 26) were added to give the user a message when taking actions which breaks the "rules" of the prototype as well as the separation rules. These notifications forces the user to confirm his or her actions when breaking said rules, which sometimes is the better option if the user knows what he is doing. The notification also works as a barrier if the user pushed a button by accident or due to a misunderstanding.

Figure 26: Notifications that reinforces the users rather than punishes them.

The notifications uses the principle of reinforcing the users by informing them of what is going on, rather than punishing them for their actions as discussed in section 6.4.1. In addition, the notifications need to be short and their meanings clear as time is of the essence during the simulation which cannot be paused.

Lastly, there was added a start screen (Figure 27) as well as a statistics screen (Figure 28) for the subsequent usability testing. The start screen displays a logo together with a button which starts the simulation when pressed. This makes it easier to conduct the usability testing as with the previous version, the simulation started instantly after the program was launched. The statistics screen display the performance of the user throughtout the simulation by listing the number of aircraft which departed on time, departed early and departed late. This give an indication of how well the users are doing and makes it easier to compare and analyze the results.

Figure 27: Start Screen. The button prevents the simulation to start instantly when the program is launched.

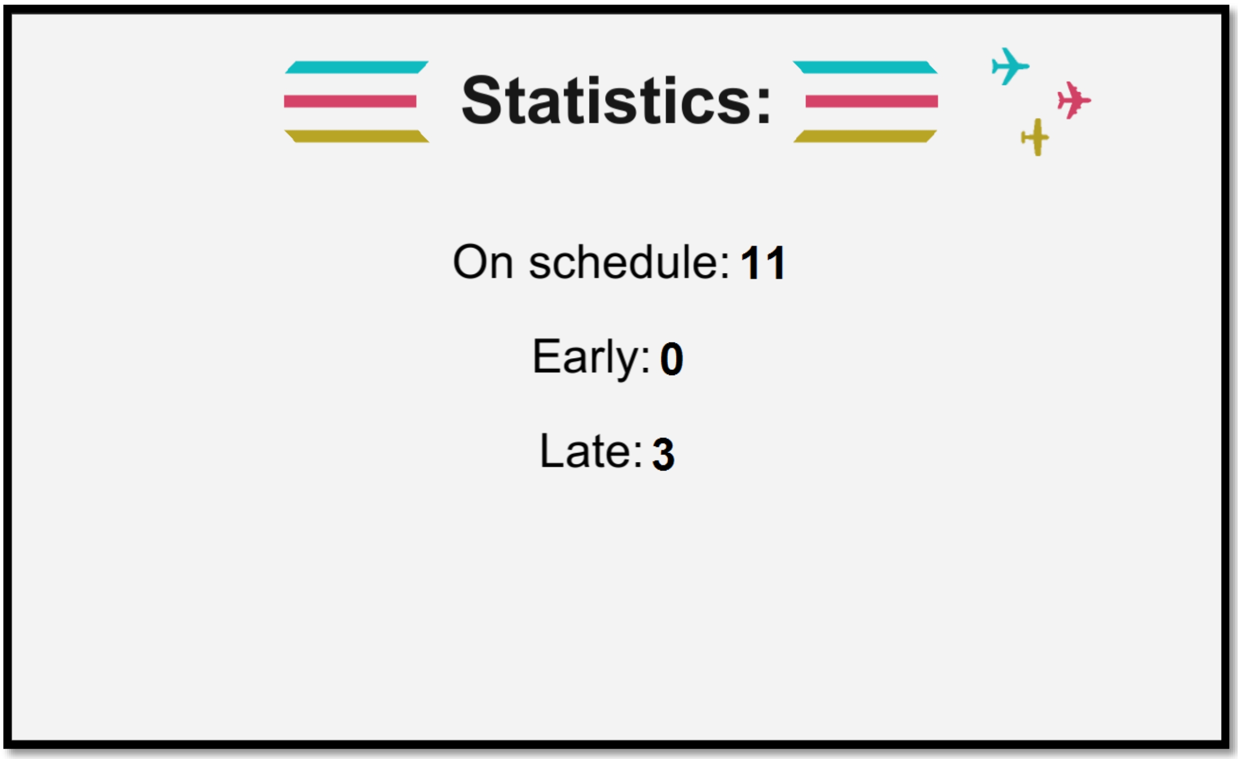


Figure 28: End of simulation statistics screen displaying how well the user performed.

8 Design challenges

This chapter will describe some of the challenges that was recognized during the design and development of the tool as well as what was done in order to address them. Section 8.1 and 8.2 will describe challenges of a more technical nature, while 8.3, 8.4 and 8.5 will describe the more design oriented challenges.

8.1 Support for multiple resolutions

This tool is intended to be something that ATM trainees can use with their own equipment, wherever and when they would like to. As a lot of the equipment used will have different display resolutions, a challenge arises to create this tool in a way that would support a large variety of displays. This problem was uncovered as the development of the tool took place on a desktop computer with a resolution of 1920 x 1200 pixels and was later moved to a laptop computer with a much smaller resolution of 1366 x 768 pixels. This resulted in menus, text and other graphical elements being compressed on the smaller screen, creating a very clunky menu and a cluttered appearance.

During development, it became apparent that the problem was rooted in the way the size and the position of GUI elements were defined. Up to this point, the elements were defined by specific sizes and positions within the scene itself. The solution was to remove the specific numbers defining sizes and replacing them by using anchor points together with stretching and padding in order for the GUI elements to adapt to the current resolution. The anchor defines which part of the screen the element should be attached to, for instance upper left corner. The padding can then be set in amount of pixels from this anchor an element should be placed, while a horizontal or vertical stretch means that the element will cover the entire display in either direction regardless of the resolution. Multiple GUI elements can be placed within others together with *layout groups* in order to arrange them relative to each other.

For practical reasons, very small screens such as smartphones will still appear as cluttered as elements have certain minimum values to make them readable. Development for smartphone sized displays has been avoided as they are deemed too small to fit the purpose of the tool.

8.2 Interface / user controls

As the SESAR project utilises an algorithm that automatically generates optimal or close-to-optimal solutions for when aircraft should go off-block and setting up sequencing for which aircraft to go before others based on rules of separation. There is a question of how much control the system should leave to the user and how much control should be given to the system itself. In other words, figuring out which Sheridan Verplanck Level is the optimal one in an ATM situation. In the case of the tool it is a challenge to figure out which SVL that better promotes learning. It is likely that a high level of automation won't be good for educational purposes as the automated system will do most actions without any operator input. Another challenge when developing the tool was to define which actions that should be available to the user. In a scenario reflecting the real world, a team of ATCOs should have the possibility to control any aspects of ATC such as issuing clearances for final approach, instructing airplanes to enter a holding stack, delaying departures, and so on. Dealing with all of these aspects of ATC would be too difficult and time consuming for this project and has been limited to only issuing orders for aircraft to perform take-offs. The main purpose of the tool is to see how visualisations are able to promote learning in ATC scenarios. Developing a very advanced and intricate prototype does not only require a very long time, it also makes it much harder to identify and isolate which elements of the prototype that works from those who do not. When it comes to prototyping, less advanced prototypes are less likely to shape the thoughts of any person who tests it, thus the feedback is likely to be more creative (Brandt, 2007)

8.3 Icon/Symbol usage

Icons usually depict familiar symbols, images which act as metaphors of real world objects and activities (Koutsourelakis and Chorianopoulos, 2010), and are used extensively in modern user interface design. In fact it has become one of the most universal aspects in graphical user interfaces of many types of software (Barr et al., 2003). If you have a smartphone, a quick look at its screen will likely reveal icons for remaining battery life, operator signal strength, Wi-Fi signal strength, with a high possibility other icons present. Icons, if designed properly, can depict a lot of information with the use of small screen space. In general, understanding of an icon relates to the users previous experience with similar or the same icons (Bedford, 2014). In addition, an icon can only be described as successful if the users recognize the meaning behind the icons matching the designers intent (Barr et al., 2003).

Bedford (2014) mentions a number of benefits regarding the use of icons:

- They can be compact and take little screen space, easy to arrange.
- Icons can be easily and quickly recognized.
- They are universal across languages if cultural differences are accounted for.
- Icons can be, if aesthetically done, visually pleasing.
- Icons can be used to support a product family or a brand.

However, as there are potential benefits to icons, there are also drawbacks. One of those drawbacks is that there are very few icons that are universally accepted such as a floppy disk as an icon for saving files and the magnifying glass for searching. Bedford (2014) therefore suggests that every icon should include a text label to emphasize its meaning. However, in most software, icons exist within the context of the application itself, usually alongside other icons which can help explain the meaning of icons (Barr et al., 2003). In the case of the learning tool developed simultaneously as this is written, the application area is that of an airport and the icons depicting airplanes are pretty much self-explanatory.

To test the initial designs for the icons to be used in the learning tool, we conducted a so called *Icon intuitiveness* test where icons are to be presented for the participants without any text label (Nielsen, 1995b). This test was conducted with five participants as suggested, all of which were shown the different icons and was told to write down what they thought it meant.

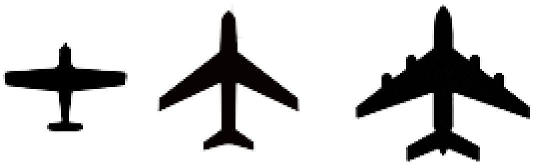


Figure 29: Icons used to depict different sizes of aircraft.

When the above icons were showed to the participants' one by one, most of the answers were just that of "airplane" and in one case for the leftmost icon it was described as "propeller aircraft". However, when all three was presented side by side, the common answer was "aircraft in different sizes" which is exactly the meaning behind their design.

Figure 30: Stop signs. First iteration (A). Second iteration (B).

In the above figure, icon A, when presented to the participants produced the feedback of something being illegal and was compared to similar signs with dogs and skateboards underneath the red crossing bar It was also said that it closely resembles a norwegian prohibited parking sign, only that it has a blue background. The intention behind the icon was that the air traffic controller would know that there is an incoming aircraft closing in on the airport and that no other aircraft should enter the runway. We decided to present another existing traffic sign, as they have an already established meaning known to most of the general public. Icon B was generally understood by the participants, but it was perceived as more scary in the sense that it would feel like if they were doing something wrong if that symbol would suddenly appear on their screen while working on their computers. The intention was for the symbol to be a warning sign more than rather than something which appears as punishing, therefore icon A was kept as the participants found it more comfortable of the two.

Barr, Noble et.al (2003) focuses on the work of Charles Sanders Pierce and present three different types of signs: iconic signs, indexical signs and symbolic signs. Iconic sign is when it relates to an object through resemblance, such as the aircraft icons in Figure 29 above. Indexical signs happen when icons depict the cause of a desired action, such as an icon of a printer printing paper. A symbolic sign relates to an object or action based on resemblance (and text) in the same way that many roadsigns does.

8.3.1 Icon language

Even if a set of icons presents a clear meaning to you, they might produce other meanings for other people than yourself. As mentioned earlier, understanding of an icon is based on previous experiences. There have been done research on how cultural differences may impact the understanding of design in software applications (Kim and Lee, 2005; Smith and Salvendy, 2001) as well as research on how age impacts understanding of icons (Koutsourelakis and Chorianopoulos, 2010). Although, this research shows that consideration in universially translatable icons should be emphasized, the aviation industry largely uses the same equipment and operates within very similar contexts. In the case of the safe-critical nature of ATM, it might be worth to look into cultural differences regarding the interpretation of icons and the understanring og user interfaces when the SESAR project is further in its development of the new ATM systems.

8.4 Testing

There will be a limited time to test the prototype. To not bore the participants, and also to have something to measure, the air traffic scenario depicted in the prototype is not based on any real air traffic patterns and is just there to measure if the participants are learning anything from interacting with the prototype. The scenario that was used during evaluation #2 (chapter 10) included 6 arriving aircraft and 14 departing ones, some of which were also arriving before scheduled to depart again. The scenario was designed to last for 20 minutes to fit into a scheduled 30 minute session.

8.5 Colour usage

Colours are an important aspect of any design, as colours often are associated with certain values or expectations. In general, colours can be divided into three main categories for describing them; warm, cool and neutral. Warm colours are often associated with intensity and energy. Cool colours are often related to calmness, nature and stability. While the neutral colour of black relates to elegance and white to purity (Bear, 2015; Chapman, 2010; Levkowitz, 1997). In the tool, we wanted the learning tool to promote a relaxed and stable atmosphere and decided for the main colour scheme being blue. Blue symbolizes depth, trust, confidence and intelligence amongst others (Color Wheel Pro, n.d.), of which trust and confidence in a system that is made to promote learning is very important.

It is suggested to use three main colours when deciding upon a colour scheme in the ratio of 60% main colour, 30% secondary colour and 10% accent colour (Tigercolor, n.d.). There are also some basic formulas that addresses how to reach harmonious colours, suggesting to base the design on analogous colours or complementary colours. Analogous colours mean any three colours that are next to each other in a colour wheel with twelve colours. Complementary colours are colours that are found on the opposite side in a colours wheel (Figure 31) (Colormatters, n.d.)

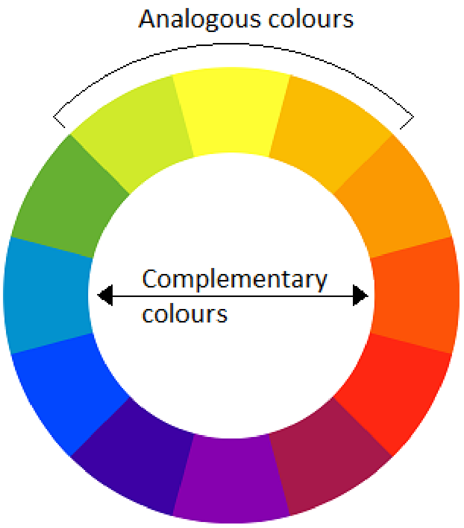


Figure 31: Twelve-colour colour wheel

For the RSMLT, we decided on using an analogous colour scheme consisting of mainly three colours ranging between blue and teal to promote a sense of professionalism.

In a combination with the “cool” blue colour scheme, intense and warm colours such as yellow and red are used to highlight objects of interest in the tool. In colour and visual search, colours that stand out from the others allow for preattentive vision (Figure 32). This means that the colours attract the users vision without taking any effort by the user to search for it, and at the same time it won’t interfere with any cognitive task currently at hand (Levkowitz, 1997). In the RSMLT, yellow is used for highlighting a selected aircraft and its related on-screen components. The reason for using yellow is that it is a very bright and visible colour and is often used on reflective vests, emergency vehicles and signs. Red are used to issue warnings about incoming aircraft and that extra caution is needed. Red is often a colour that is related to danger (Bear, 2015) and in traffic, red lights usually means to stop.

Figure 32: Preattentive colour features (Levkowitz, 1997)

As one can see from the figure above, colours can both distract (left) and attract (right) visual attention from the user. The colours yellow and red were also considering the level of contrast between them and the blue and teal main colours.

Another challenge that came into mind was to select colours that the colour blind does not have a problem telling apart from one another. However after some research on the requirements of becoming an ATM controller, it was discovered that an applicant at the time of writing is required to have normal colour vision (Avinor, n.d.).

9 Prototype Evaluation #1

This chapter will describe the process for evaluating the first version of the high-fidelity prototype as described in section 7.4. In section 4.3.1 it was mentioned three different types of usability testing, automated testing, expert-reviews and testing with real users. Of these three, it was decided to use interface experts in an early evaluation as this is a good way to identify the biggest design flaws and is generally suggested to be done before usability testing sessions with real users (Lazar, Feng et al. 2010). This way, subsequent usability testing with actual users allows for more accurate data to be gathered about usefulness of the system by limiting the amount of interruption caused by design flaws. This chapter will describe the planning of as well as the conduction of the expert testing before finally presenting the results.

9.1 Expert testing

This particular expert testing session was designed as a cognitive review (section 4.3.1.3) and, included a presentation as well as a set number of tasks taking the experts through some of the key elements of the prototype. The presentation was divided into three main topics;

1) A very brief presentation of the SESAR project.
2) A short explanation of separation rules, what they are and why they are important.
3) An overview of the prototype, explaining its purpose and user interface.

The presentation was intended to give the participants some understanding about the context of the prototype and its use-situation, hopefully enabling the participants to give more detailed feedback during the review. The tasks themselves (Appendix D) was divided into three separate sets, each of the set placing emphasis on different aspects of the prototype with the intention of making it easier to identify why and where, if any mistakes would occur;

- User task 1: general understanding of the interface, identifying aircrafts.
- User task 2: understanding of the separation rules as defined in the prototype.
- User task 3: arriving and departing aircraft.

In addition to the user tasks, there was also included an extra task describing two examples of visual feedback indicating that a button has been activated. The intent of this task was to identify which solution the experts found the most intuitive and why.

Two pages containing tables was included to each of the three user tasks for the experts to write down any comments or problems related to a specific sub-task or question within each set. The experts were also asked to rate the problems they identified by using the severity scale introduced by Nielsen (1995c) . The scale uses the numbers 0-4 to describe the level of severity where 0 are the most harmless errors and 4 are the most critical ones:

0. = I don't agree that this is a usability problem at all.
1. = Cosmetic problem only, does not need fixing unless time allows it.
2. = Minor usability problem: low priority problem.
3. = Major usability problem: important to fix, high priority.
4. = Usability disaster: this must be fixed, highest priority.

Preparing the material for the cognitive review proved a challenge as the prototype is a single screen application with subtle changes on the screen in contrast to many other applications where pressing a button changes screen entirely. The challenge was to create a set of linear screenshots with an attached set of tasks. The tasks were intended to be meaningful in a way that experts would find the flow between tasks and screenshots a natural progress throughout the session. The initial version of the tasks and screenshots were presented to, and tested with a co-student within interaction design in an effort to improve it before the expert review. Some issues were discovered and led to reworking some of the tasks as well as an inclusion of additional screenshots. The main purpose of the extra screenshots was to indicate that certain actions had been carried out, intended to give feedback to the experts.

The main problem was to create a set of screenshots naturally transitioning from one to the next by looking at the corresponding tasks. As the intention was for the participants to go through the set of tasks themselves without the tasks explicitly stating which screenshots to use, the challenge was to design user tasks and screenshots that implicitly tell the experts when to use the next screenshot in sequence.

9.1.1 Conducting the expert testing

As stated earlier in section 4.3.1, one should use as many participants as possible when the systems area of application is safe-critical. ATM is an area most people would agree to be safe-critical, but with the nature of this thesis and its limited scope and the fact that the tool created is just an experimental prototype, having a very high number of participants is not that important. Normally, finding experts who are willing and have the time to be a part of usability testing can be difficult and costly. Fortunately, through SINTEF, six experts was recruited, of whom five were HCI-experts. One expert had no prior HCI experience, but had extensive knowledge about ATM research.

The expert testing took place at SINTEF's offices in Oslo for ease of convenience and the fact that all of the invited experts were SINTEF employees. All of the six recruited experts showed up for the session. Naturally, the experts have different backgrounds as well as differentiating levels of experience regarding HCI and ATM research. Each expert was asked to answer the following questions:

Q1: What is your highest level of education?
Q2: How many years of work experience do you have?
Q3: How many years of experience within the field of HCI do you have?
Q4: How many years of experience within the field of ATM do you have?

The level of education varied between Candidatus scientiarum, MSc and PhD. Some in other areas than software engineering. The experts' work experience ranged from 4 to 30 years with an average experience of 15.5 years with the median being 25 years of work experience. When it comes to experience from HCI, it ranges between 0 and 30 years, the average being 11 years of an experience and a median of 7 years. In experience with ATM, the results ranged from 0 to 6 years of experience with the average being 2.5 years and the median also being 2.5 years. Looking at these numbers, we can see that the experts have very different level of education as well as experience within the fields that was queried. Having experts with different levels of experience can be regarded as a strong point as they are likely to have different perspectives and expectations concerning interfaces. Overall, the experts are very experienced in HCI while somewhat less experienced with ATM.

During the presentation, there were some questions further enquiring about some details about the prototype. These were mostly about the separation rules and its relation to the sequence timer. The presentation lasted for about 15 minutes of the scheduled 60 minute duration of the expert testing.

The remaining 45 minutes was used by the experts to solve the set of tasks that was handed out. Some of the experts expressed confusion regarding which screenshot which should be used in order to solve their current task. Thus, going through the tasks required some assistance from the facilitator. An issue discovered during testing was that the printed screenshots were in black and white. The lack of colours made some interface elements in the screenshots hard to distinguish from another as the grey tones appeared to be very similar. Some of the experts also failed to understand the purpose of the additional question included at the end of the set of tasks. The question proved not being as self-explanatory as thought, thus, more explanation was required.

Picture 1: Ongoing expert-review / Cognitive walkthrough.

After all the experts had completed all three user-tasks, there was a short discussion where it became apparent that there was some confusion regarding the formulations in some of the tasks. Reflecting on the expert testing, there were quite a few issues that could have been improved upon, both tasks and presentation related issues. Regardless, the session were deemed as successful as it generated a lot of creative feedback from the experts.

9.1.2 Expert-testing results and analysis

The experts reported a total of 28 problems identified in the course of the expert-review session. Of the problems found, 47% was declared as critical (3 or 4 on the severity scale), while the remaining 53% was identified as less important usability issues (2 or less on the severity scale) as can be seen in Figure 33.

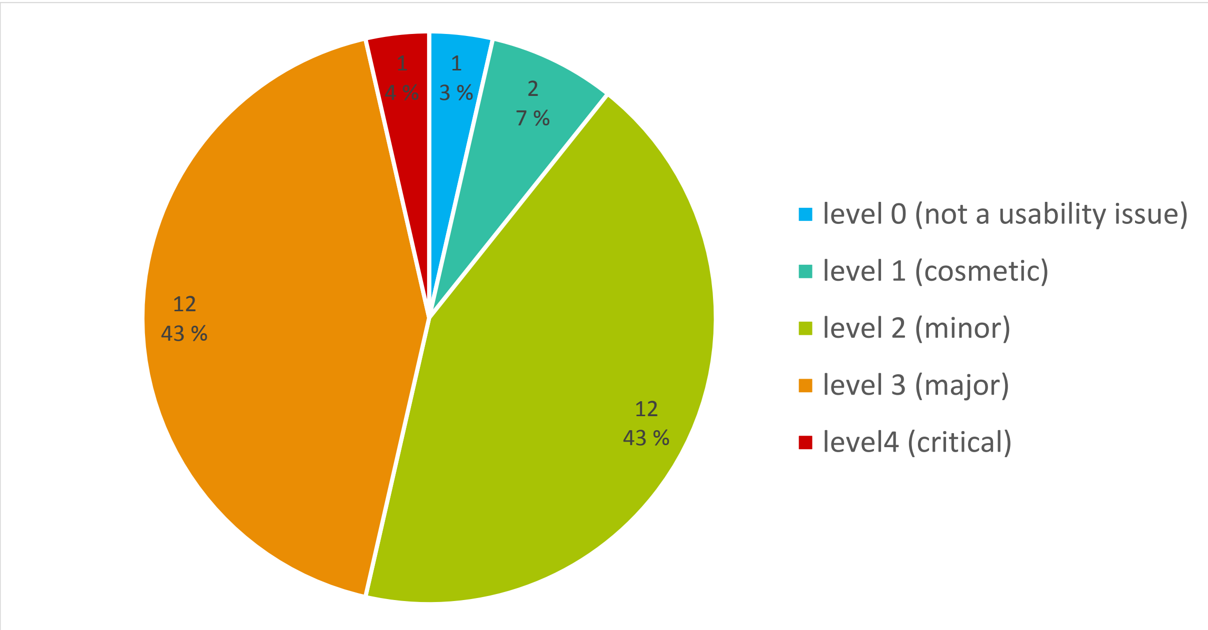


Figure 33: Pie chart showing distribution of problems identified by their level of severity.

In order to analyse the results, a technique called *open coding* was used. Open coding involves deciphering data by breaking it down analytically. Open coding is normally being used to analyse phenomena that occurs during qualitative research by labelling the collected data when comparing events, actions and interactions (Corbin and Strauss, 1990). Using open coding in order to analyse the results from the expert-review session helps to identify the underlying cause of the problems that was reported. A typical approach to open coding is to ask the question "what is this really about?" In the same manner, each of the problem descriptions that was written down during the review session was analysed by asking the question "What is really causing this to be a problem?" Through this process, six categories were created in order to contain all the 28 problems that had been identified. Following is a list of the created categories together with a short description:

- ***Uncertainty of where to look for the needed information.***
  When the experts were struggling to understand which interface component should be looked at to complete some of the tasks.

- ***Difficulty to understand interface components.***
  When the experts identified the correct interface component, but failed to understand its purpose and/or meaning.

- ***Mismatch between interface components.***
  When experts were expressed confusion by similar components not sharing some of the same characteristics.

- ***Make connections between interface components more obvious.***
  When experts reported issues whose underlying cause is failure to recognize how some interface components relate to each other.

- ***Other interface problems.***
  When the experts identified other interface problems such as button placement, lack of colour, missing time units etc.

In addition to the five categories above describing usability errors, there is another sixth category which was made to include problems not being usability issues, but was reported as a problem nonetheless by the experts and therefore should be included in the analysis:

- ***Problems with the task formulation.***
  When the experts reported problems with the actual formulation of tasks rather than the actual task itself were put in this category.

Many of the problems identified by the experts were the same. However, by adding the weight of each identification further stresses the seriousness of the issue. If an issue is blatantly visible to several of the testers, its importance in fixing should increase.

As can be seen in Table 7 and Figure 34 below, the most important categories where the most problems were identified are uncertainty of where to look (**a)**, difficulties with understanding (**b)** and other interface problems (**g)**. As these three categories have 10, 6 and 6 reported problems respectably, the three categories contain a total of 22 out of 28 errors that were identified through the session. Based on the identified problems alone, most of the future effort in evolving the prototype should be emphasised on these categories. Special emphasis should be placed on category **a**, as this category contains by far the largest amount of errors as well as the most severe errors with a total of 9 out of 13 errors with a severity rating of 3 or higher.

Table 7: Error categories and frequencies of problems within each category. Numbers below each level of severity shows the amount of errors identified on that level within the category that appears on the same line.

	Severity of reported issues → Categories of errors ↓	level 0	level 1	level 2	level 3	level 4	Sum
a	Uncertainty of where to look for the needed information			1	8	1	10
b	Difficult to understand interface component			3	3		6
c	Mismatch between interface components		1				1
d	Make interface components connections more obvious			3			3
e	Group components of similar function			1			1
f	Problems regarding the tasks themselves			1			1
g	Other technical	1	1	3	1		6
	Sum	1	2	12	12	1	28

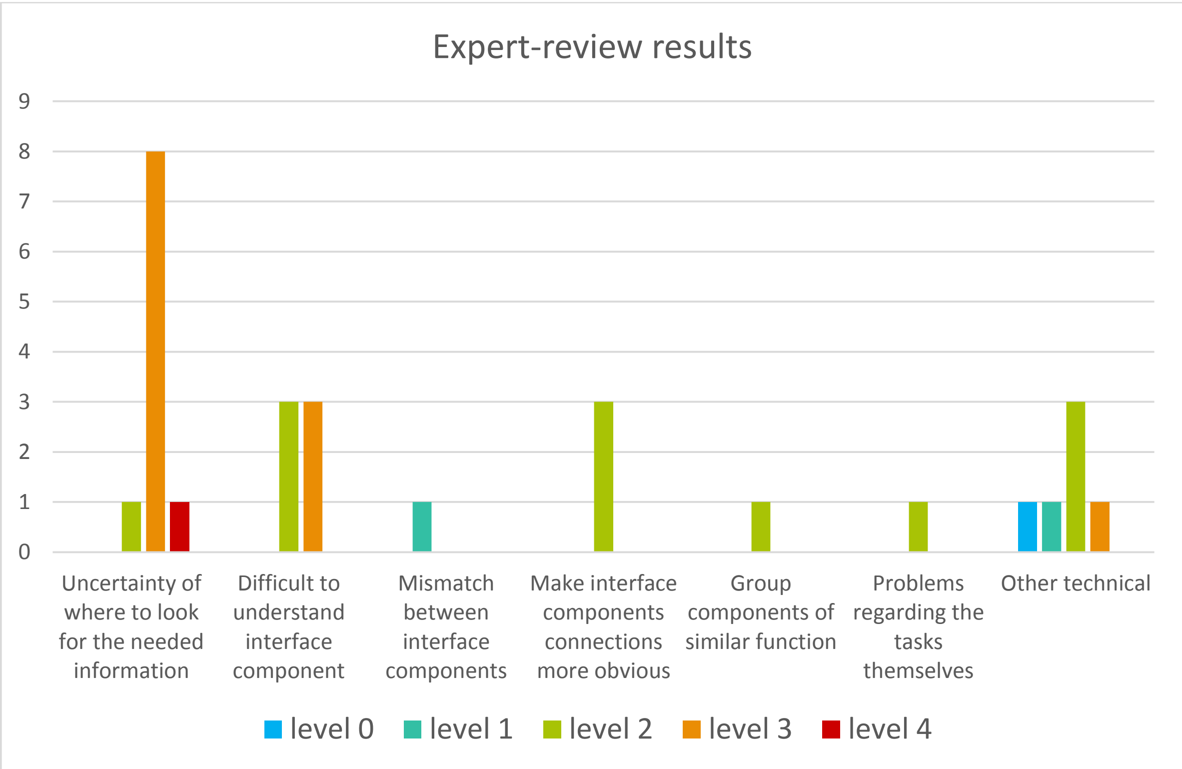


Figure 34: Graph showing problem categories and the frequencies of problems in the different levels of severity.

Having identified the most common issues called for investigation of when and where they occur. Mentioned earlier in this chapter, in section 9.1, each set of user tasks had different areas of the prototype where emphasis was placed:

- User task 1: general understanding of the interface, identifying aircrafts.
- User task 2: understanding of the separation rules as defined in the prototype.
- User task 3: arriving and departing aircraft.

By sorting the problems by which user task they were identified from, we were able to gain some knowledge about which parts of the prototypes' interface that had the least amount of problems and which part that was most problem infested. Almost half of the problems, 46% were identified during set 2, whereas set 1 and 3 got an even amount of 25% and 29% (Figure 35). Problems were also identified based on categories within each set of tasks (Figure 36). The combined results from these two figures revealed that the second set was the most problematic one, containing more identified issues in all the categories except *other technical* and the *mismatch* categories. These findings indicated that most critical issue regarding the prototype was related to understanding of the separation rules, thus emphasis should be placed to resolve the issues related to this area.

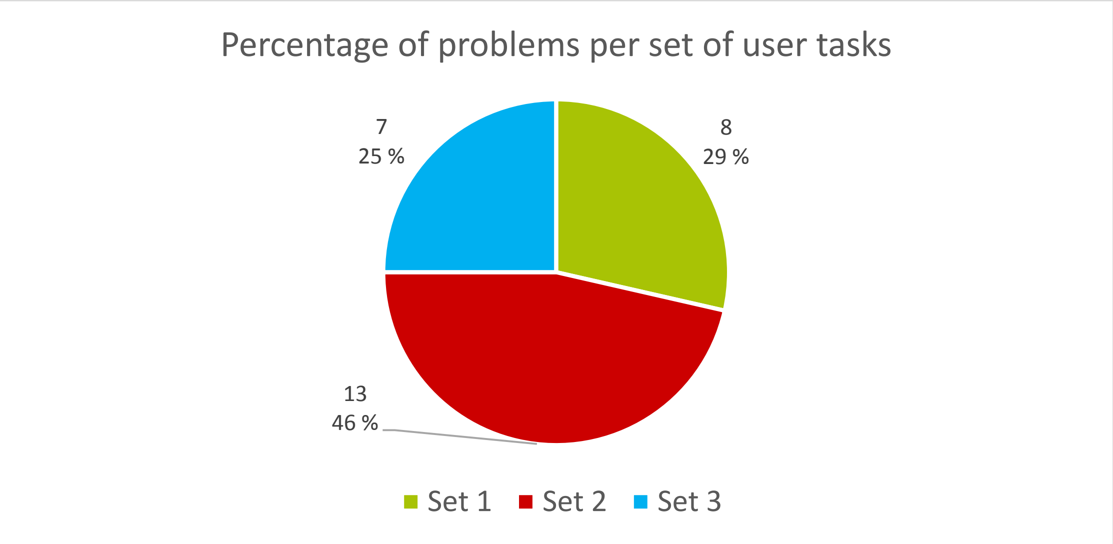


Figure 35: Pie chart showing error distribution among the different sets of user tasks.

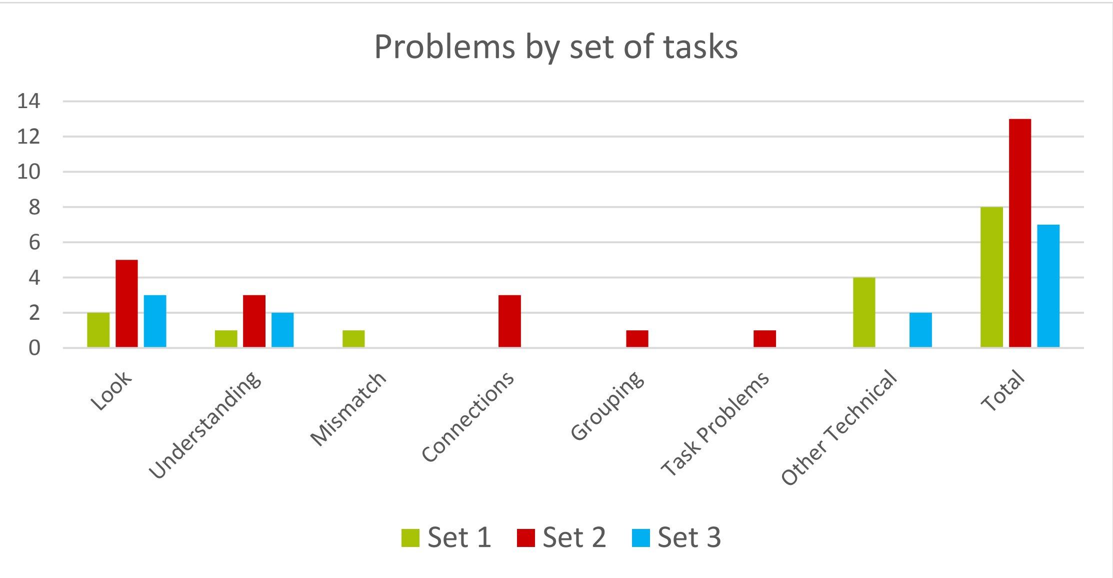


Figure 36: Graph showing distribution of identified problems within each set of user tasks.

As ascertained earlier in this thesis (section 4.3.1.3), one should be careful when analysing the results of an expert-review as what might appear as an issue to an interface expert, might not be an issue to a real user. From the debriefing session after the tasks were completed, it was made clear that several of the experts were confused about the tasks and what they were asked to do, as was also made clear from the results. The most noteworthy interface element that were discussed was the sequence timer and what its purpose was. One could argue that real users would get more thorough training than the experts, which is a valid assumption. However, the prototype should strive to be as user friendly as possible and intuitively tell the user what the different elements in the prototype do and how they work together in order to facilitate learning.

During the debriefing, the experts expressed that the tasks themselves perhaps should have been refined more in order to make them more understandable from their point of view as someone who has not been involved with the prototype before. A difficulty towards understanding a task may have led to situations where problems were perceived, but would not appear as a problem if the task was fully understood beforehand. Overall, the expert testing resulted in uncovering many issues that should be considered when continuing the development of the prototype. The main area of focus should be to design a more intuitive GUI, both in appearance and placement of components in order to make their functions more obvious and understandable. Special focus should be placed on improving the visualisation of separation rules as this was the tasks related to them were the most confusing for the experts to figure out.

9.1.2.1 Analysis of the additional task

The additional task was for the experts to look at two separate solutions of what a button should look like when activating a menu. Alternative 1 changed the colour of the button to indicate that it was active while alternative 2 change the text from "Show separation rules" to "Hide separation rules" while retaining the same colour in both states.

Results for alternative 1:

- The button shows toggling on and off.
- The change in colour shows the selected function.

Results for alternative 2:

- This is better if the goal is to show that the separation rules are visible.
- It tells you what you can do by pressing the button, maybe the two alternatives could be combined?
- Changes the state more visibly.
- You click what the button tells you.

The results indicated that two out of six experts preferred alternative 1, while the remaining four preferred alternative 2. Considering that there are only six experts, the sample size is too low to fully conclude that alternative 2 really is more intuitive than alternative 1. However, one of the experts suggested that the alternatives could be combined in order to create a third alternative. This last alternative should be explored to see if the combination of alternative 1 and 2 makes up for a more intuitive solution or doing the contrary, making it appear as confusing.

10 Prototype Evaluation #2

This chapter describes the second evaluation of the prototype, a usability test. The expert-review raised several issues with the prototype, some of which was addressed in the next iteration of the prototype (section 7.5). The usability testing session was conducted with users representing the target group (ATCOs) rather than actual ATCO students with the goal of testing aspects in usability of the final version of the prototype as presented in chapter **Error! Reference source not found.**.

10.1 Recruiting testers

Naturally, before actually conducting the usability testing, willing testers needed to be recruited. Students, while not specifically students for ATC were deemed adequate as the tool is intended for use as a part of a learning environment. Due to unforeseen circumstances, the tool was not ready for usability testing as early as anticipated. This made it hard to recruit students at the university as most are gone for the summer. In order to reach the suggested number of minimum five users (section 4.3.1), some participants were recruited from outside the university. Of the recruited participants, three are MSc students in interaction design, one has a BCs in animation production and the last one has a BCs in construction engineering. This means that the users have very different experiences and background, likely leading to more diverse feedback compared to having all testers sharing a similar background.

10.2 Conducting the usability testing

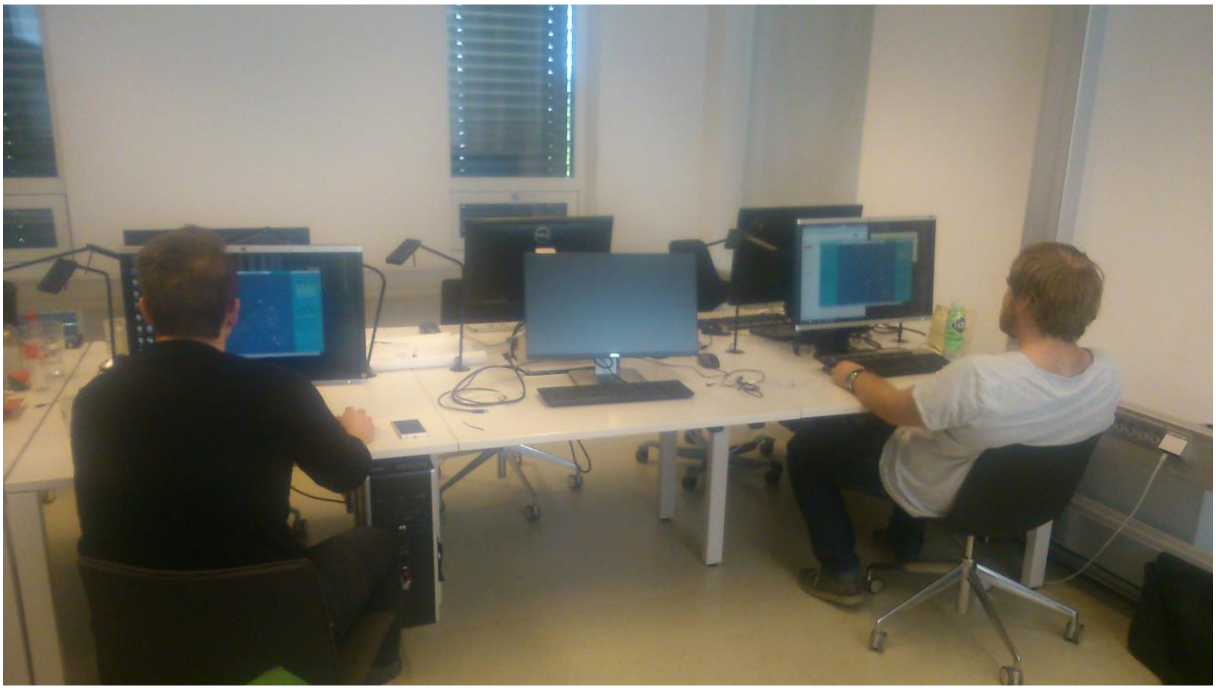
Due to recruiting users from different places and limited availability for everyone to travel to the same place, the usability tests took place at different locations using slightly different methods. Lazar, Feng et al. (2010) recognizes that there are benefits and disadvantages to different locations. In short, usability testing in laboratories requires the participant to travel, and the laboratory setting might make the participant to feel uncomfortable. Usability testing at the workplace or home is a more natural setting for the participant, but may bring other challenges such technical issues by setting up equipment at a new location. Lastly, remote usability testing can be used when a meeting is not feasible, however, this makes it harder to pick up on non-verbal cues (Lazar et al., 2010). Unlike the literature, the usability testing in this project took shape in different forms on the same product, due to the availability of participants. It was emphasised to require as little as possible from the participants in the form of travelling due to the uncertainty of when the tool would be deemed ready for testing. This meant that the usability testing in this project took place at the workplace, people’s homes and by remote usability testing. The different usability testing sessions taking place on specific locations will be described in more detail later in this chapter.

The usability testing was designed as a combination of *formative* and *summative* testing. Formative testing is normally employed early in the process and focuses on generating qualitative feedback and uses and exploratory approach to discover problems. Summative testing on the other hand is more task-oriented and is used to gather quantitative measurements and metrics (Lazar et al., 2010). The decision to combine these two types of testing is due to the state of the prototype, while it is a high-fidelity and fully functional prototype as is typical for summative testing. It is also a really early prototype in the greater sense of designing a complete learning experience for ATCOs in training.

Regardless of where the usability testing were undertaken, all the participants were required to sign a consent form (Appendix A) before testing commenced. The participants were made aware of that they were not being tested, but they were the ones testing the product. They were also encouraged to think aloud, making it easier to observe their thought process. All the participants were given the same task, which was to complete the scenario in the tool by issuing take-off orders to all the aircraft in the departure list while respecting the separation rules as described within the tool (section 5.2). After the participants had completed the scenario, they were asked to fill out a form (Appendix E) containing four statements where the participants were asked to fill in their level of agreement on a Likert-type. In addition there were two more questions asking the participants to write down two things they liked about the tool and two things that they did not like as suggested in Spool (2006). Spool also suggest taking notice of how quick the participants are to answer these questions as it can be an indicator of how much weight the answer should be given. If the participant write down their answers immediately, the answer should be given more weight when compared to answers where the participant must think for a while before writing it down (Spool, 2006). After the paperwork has been filled out, there was time for a short discussion where the participants were asked to express their feelings towards the prototype and suggest any improvements they could think of.

10.2.1 Usability testing at the workplace

This part of the usability testing concerns testing with two students at the institute of informatics which in every sense is their workplace. The usability testing was conducted with both of the students at the same time (Picture 2) with the hopes of sparking a group discussion at the end of the session. The participants played the scenario on university computers as the RSMLT software could be loaded from USB sticks prepared in advance.



Picture 2: Usability testing at the workplace.

Being encouraged to think aloud sparked a conversation between the participants as they explored the interface and tool in the following form:

> **Participant A**: I can send aircraft X now, but I think I will wait.
> **Participant B**: I sent Y before X because of the separation rules.
> **Participant A**: Did you? That means Y is departing before scheduled time.
> **Participant B**: Yes, but there is no indicator saying that the aircraft is not ready.

This conversation indicates that the participants have different perceptions of what they are supposed to do with the tool and one participant's opinions might influence the others by thinking the other person is right and he is wrong, even though that might not be the case. During the testing, the participants got more and more inclined towards ignoring the notifications (Figure 26) as they saw no reason not to do so. (Why is there an option to ignore the message if it's not meant to be used?). The statistics screen (Figure 28) was bugged and did not work as intended (more details in section 10.3), but it was noticed that a large portion of their departed aircraft was sent earlier than scheduled.

10.2.2 Usability testing at home

The participants with degrees in animation production and construction engineering had the usability testing session taking place at their respective homes. Like with the participants in the workplace, they received USB sticks containing the RSMLT to use on their own preferred computer. The animation production participant opted to sit at his stationary computer on his work desk while the construction engineer used his laptop while seated in a sofa. Both of the participants here had a different approach to the tool than the designers at the workplace. They both respected the separation rules above getting aircraft departed on time as they assumed safety is the more important of the two, although they had different approaches to departing aircraft. The animator pressed the take-off button whenever an aircraft's take-off time was reached and only looked at the separation timer if there was a notification box popping up indicating a breach of separation rules. He would then press cancel and wait for the separation timer for that specific aircraft type to reach zero. The construction engineer on the other hand marked the next airplane in the list, then consulting the incoming aircraft list for conflicts, then looked at the current time and lastly the separation timer. He then patiently waited for the current time to match the timeslot of the aircraft and/or for the separation timer to reach zero before pressing the take-off button.

10.2.3 Remote usability testing

The remote usability testing were conducted with another MSc student in interaction design as we could not find a timeslot available for both of us within a reasonable time. The testing were asynchronous, meaning that the participant did the tasks when he had time to do it and there were no communication during the actual testing. The tester received a compressed file containing the tool, the consent form, the follow-up questions and a readme file containing a short description of the task at hand. As the facilitator is not normally supposed to interfere while testing (Lazar et al., 2010), the participant would need some direction and sense of purpose without revealing too much detailed information in order to play through the scenario and to give proper feedback.

10.3 Technical issues

During the actual testing, there appeared to be some technical issues with the tool which was not yet discovered. When launching the program, unity offers the options to either select a resolution to play in windowed mode or to play the program in full-screen. Even though, one of the requirements of the prototype is to handle multiple resolutions and was tested with multiple resolutions, it still appeared distorted in some aspect ratios, stretching some interface elements. Particularly affected by this was the separation rules bar (Figure 12). The participants were then asked to start the tool in 1366x768 pixels to avoid this issue.

Another issue which became apparent during testing was the appearance of some unexpected aircraft landing at the airport which was not listed in the incoming aircraft panel, but still triggered the incoming aircraft warning, thus blocking all take-off actions until the aircraft had landed. Although the unexpected aircraft did not interfere with the schedule too much, they still made the participants confused at what was going on.

The most important technical bug discovered while testing was that the statistics screen (Figure 28) did not work, although it had worked earlier when tested. The intentions behind introducing the statistics screen was to give the users of the tool something which gave them an indication of how well they performed, but it was also intended to be a way of measuring quantitative feedback as is suggested in summative user testing. The statistics screen ended up counting all departures as being "too early" when it was observed that this was not the case.

10.4 Findings

This section will discuss the findings from the notes taken during the simulation, the post-test questionnaire and the following debriefing and discussion part of the session. Due to the nature of the usability test, the findings are both of the formative and the summative kind. The first part of this section will focus on the written feedback from the participants, the questionnaire, followed by an analysis of the notes taken during the simulation and the debriefing.

10.4.1 Findings from post-test questionnaire

For the ease of reading, the questions from the post-usability test questionnaire are translated from Norwegian and (Appendix E) listed beneath. Firstly, the participants were asked to read four statements which they were tasked to mark their level of agreement on a Likert-type scale ranging from 1 (strongly disagree/ very negative) to 6 (strongly agree/very positive). The results are presented in Figure 37 below. The statements were the following:

Q1: It was easy to gain an overview of the components in the tool
Q2: I quickly understood all the functions in the tool
Q3: The buttons were placed where I expected to find them
Q4: My overall impression of the tool is:

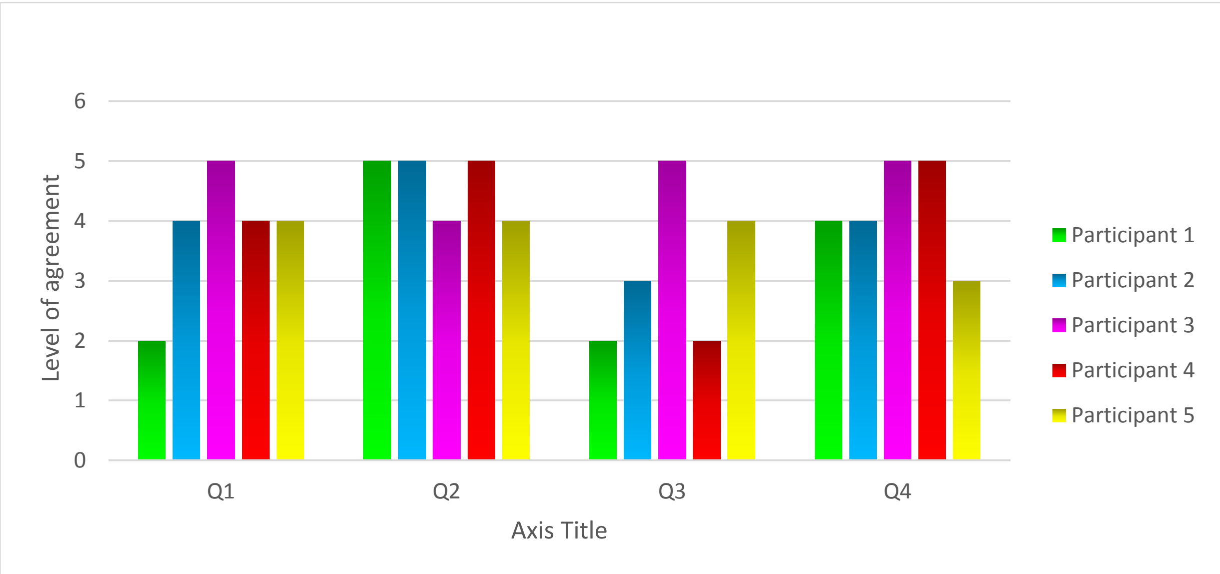


Figure 37: Results from post-usability questionnaire.

The results indicate that the participants had very differentiating opinions regarding the four statements. Overall, the answers ranged from 2 to 5, without the highest and lowest scores being used. Q1 scored an average of 3.8, indicating that the participants slightly struggled to get an overview over the different components which constitute the tool. Q2's average score of 4.6 comes as a bit of a surprise as the participants made the impression during that getting an overview of the components as difficult, yet it seems they had a fairly easy time of understanding the purpose and usage of each component. Q3 reached an average score of 3.2 which strengthens the findings of the experts during the expert-review (chapter 8), the buttons, specifically the take-off button should be moved and made more visible. Q4 got an average score of 4.2 which indicates that the overall impression of the tool was decent, although the participants were unsure on how to answer this as they were not ATCO students (more on this in section 10.4.2).

The next part of the questionnaire contained two questions where the participants were asked to write down two things they liked and two things they disliked about the tool. The gathered responses have been translated below, some of the sentences have been altered for readability reasons, but their meaning is the same.

Q5: Please write down two things you think is good about the tool.

- Animations on the planes and the feedback on the take-off part.
- A good graphical tool.
- Easy to take off/land.
- Good visual overview, not many sub-menu's.
- Intuitive.
- Soothing colours.
- Nice with a visualisation image (radar).
- It has a potential to be cool.
- It gives a nice overview of where the components are placed, and it was fairly easy to understand what to do after a while.

Q6: Please write down two things you did not like about the tool.

- The take-off button is small and could have benefited from a better placement.
- It should be possible to select aircraft from the list.
- What is the purpose of the "game" In the beginning, I did not understand where to click.
- I was sitting and waiting for aircraft to land, so I opted to depart aircraft before their schedule because I was bored
- You have to move your eyes a lot between the columns on the left and right side of the screen
- Some issues in user-friendliness
- Hard to get an overview over the time for all the incoming and departing aircraft
- Many things to keep track off, is it possible to combine them?

By looking at the responses above, it shows that the positive feedback were mostly on visual aspects of the tool. This might be rooted in multiple cases. The participants have only been interacting with the prototype for a very limited amount of time and their answers may have been influenced by what they can see visually rather than the core of functions themselves. Compared to the list of things they did not like, the answers are vaguer which is likely to do with problems being more prevalent and memorable as they encounter them. Even though participants mostly expressed specific problems, some of the answers here was that that the purpose of the tool was unclear to them and that it was boring to sit and wait. This tells us that for future testing, the exact purpose of the tool should be made clearer to the participants.

10.4.2 Findings in notes from simulation and debriefing

As the participants were encouraged to think out loud, most of them seemed to ask questions right away about certain elements in the prototype such as, what is A, B and C? It turned out that during development changing the aircraft types in the departure list from A, B and C to S, M and H (section 7.5) had been overlooked. Nevertheless, the participants quickly figured out the connection themselves without being given the answer. Two participants also found it strange that aircraft moved with constant speed regardless of whether they were flying, taking off or taxiing on the runway and said it was unrealistic. This indicates that there was a mismatch between the external and their internal representations (section 3.3), meaning that reflections between the simulation and their perceived image of the domain is taking place, supporting the creation of a cognitive schema (section 3.6.1)

All of the participants struggled with finding the button that makes the aircrafts take-off, one of them specifically said "I'll just assume a button will appear when the aircraft should depart". It was only observed that three of the participants found and used the tooltips (section 7.5) which explains the functions in more detail. However, as discussed in the previous section, the participants felt that they easily understood the purpose of the functions.

Most of the participants expressed concern towards no indicators displaying whether the aircraft was actually ready or not, as in if it has finished boarding or all the cargo has been loaded. This lead to some participants issuing take-off orders early as there was no indicators that communicated that this was a less than optimal idea besides the notification box which they ignored. A participant with earlier experience working as a baggage handler wondered if the arrivals list contained the pre-planned scheduled arrivals or if it was an estimated arrival time i.e. if the list was adjusted for delays and deviations from schedule.

All of the participants was confused that it was not possible to select aircraft in the list of departures. Regardless, the participants quickly figured which out that they could click the aircraft themselves and shortly after they figured out to find the correct location of the aircraft they were looking for by consulting the gate number in the departure list. Still they all expressed that clicking in the list would have been easier. One of the participants noticed that he made a mistake just as he dispatched an aircraft, he then tried to plan his following action in order to make up for it. This incident and how the participants figured out how to select the correct aircraft indicates that learning is taking place as the user have touched upon situation awareness level 3 – projection of future states and events (Figure 5).

During the debriefing, the participants offered a lot of suggestions which in their opinion could improve the tool. One of the ideas that occurred multiple was to place a much bigger take-off button at the bottom right corner of the screen. One of the participants also suggested for this to be colour coded based on the separation timer meaning that green would indicate that the aircraft is ready to take-off. Four out of the five participants expressed that there was a need to change the general layout of the tool as they felt the need to move their eyes across the entire screen all the time as taxing. A solution to this problem could be moving the main view (Figure 9) to one side of the display with all the functions being closer together on the remaining half. The last participant however thought the current layout with departure elements on one side and arrivals on the other made sense and were content with it.

Other suggestions included introducing another timer just like the incoming aircraft timer, but for departing aircraft. Combining the arrivals and departures list into one list with markers indicating the arrivals and departures as this would remove the need to compare times between the two lists all the time. Having some indicator that showed the effects of sending an aircraft i.e. if aircraft X is departing now, how will that affect other aircrafts schedule concerning the separation rules.

It was observed that all the participants opted to use the mouse for everything, the function buttons were untouched and so was the camera movement controls. This might be reasoned with the small and simple layout of the airfield which wholly fits within the main view. In the literature review (section 3.4) it was revealed that animation is generally a faster solution towards information retrieval compared to textual representations. This was reflected when observing the users as they spent more time to read the information in the arrivals and departures list compared to the other more visual oriented features.

10.5 Usability testing analysis

The findings from the usability testing were largely formative in the sense that it is more qualitative than quantitative. As stated earlier, the quantitative collection method implemented into to the tool in the form of post-simulation statistics did not work, thus resulting in less measureable results. It was revealed in the results from the previously conducted expert-review (chapter 8) that there was room for improvements to the user interface. The findings from this usability test suggests that while the different UI components were sorted based on arrivals and departures this time, the participants still struggled slightly to get a good overview of them. At the same time, 4 out of the 5 participants felt that they had to move their eyes all over the screen for the duration of the simulation. Considering that this simulation only lasted for 20 minutes, the interface layout needs to be reconsidered in order to make it less taxing for the more intense sessions closer resembling real life scenarios.

The usability test shows that the participants themselves felt that they understood the application for each of the components very easily. This suggests that the visualisation techniques employed in the tool works in the sense that they are easy to comprehend. This is however hard to verify as the performance statistics are missing and the questionnaire only represent their subjective perception of the tool and the scenario. On the other hand, observations showed that some aircraft were departed early which may indicate that the understanding of the components were not as good as the participants judged themselves. Again, the reason for aircraft being sent early might be due to a misunderstanding on how to use the tool as some of the participants put more weight towards aircraft not being late than the separation rules and the actual time-table.

It was identified that the participants spent more time looking for information where it was presented textually compared to the visually oriented elements. This could lead to situations where a user places too much focus on doing one task while overlooking other tasks which might be of higher importance. This phenomenon was identified as load-shedding in the literature review (section 3.4). To address this issue, the textual presentations needs to be rethought in order to make them easier to read by the users and more visual and maybe audible cues could be exploited to improve the tools adherence towards Baldonado et al. (2000)'s eight rule, the rule of attention management.

11 Discussion

Results from the expert-review and the usability test suggest that visualisation and animation may be beneficial for learning purposes within the domain of ATC. This chapter will discuss the findings from evaluations conducted during this thesis. Furthermore it will address identified limitations regarding this work.

11.1 Discussion

The results from the final evaluation conduced on the RSMLT indicated that users with no previous experience in ATM managed in some degree to interpret and understand a visual representation of separation rules. This basic understanding of separation rules was achieved by participants only after getting a brief explanation of the goal within the simulation. One of the main ideas behind the learning tool is that training should be more accessible for ATCO students, removing the necessities of using an expensive simulator and having an instructor present implying that a student is responsible for assessing his own performance.

As revealed in the literature review, Ødegård (2013) found in his work on self-assessment that users with little or no basic knowledge about what to evaluate could direct the user to falsely assess his own performance. While the tool that was developed for information visualisation and insight rather than specifically as a learning tool, he also found that students with minor existing knowledge gained less insight than more experienced students due to reduced analytical capabilities caused by not having basic knowledge about ATC processes. While the work of Ødegård (2013) focused on self-assessment and insight, these findings are highly transferable for the purpose of this thesis as insight requires a certain level of understanding and needs to be developed in a similar way that the processes of learning ideally leads to better understanding of a concept. These findings indicate that students using a tool such as RSMLT should not assess their own performance as they are on a novice level which may lead to false assessments, making the tool work against its purpose.

In order to assure that the student is improving on ATC concepts, some other form for performance evaluation needs to be in place. One option could be to have students completed simulation sessions stored on a server from which an instructor could fetch and review performances in an asynchronous fashion, meaning that the simulation and the review is being conducted at separate times. Because of the internet, this means that any instructor could potentially rate any student's work from anywhere in the world, potentially promoting flexibility for both trainer and trainee. Another option could be introducing a deep analytical algorithm embedded in the tool that enables a more detailed depiction of performance as well as indicating areas where students perform well and areas in need of attention. One of the suggestions for future work made in this thesis is to introduce an analytic post-simulation statistics screen (Figure 38).

The usability-test indicated that students with no prior knowledge were able to learn basics of air traffic separation done for aircraft departing from the airport. However, it is difficult to assess whether the knowledge gained here is transferrable and supportive to already existing methods for learning ATC such as the use of simulators. For tools like the RSMLT to have any value, learning gained from the tool must be transferrable and applicable for actual ATC utilizing real ATM systems. As it has been discussed by other authors, a longitudinal study is necessary in order to accurately measure the effects of visualisation (Schneiderman and Plaisant, 2006; Ødegård, 2013). Limitations on this thesis meant that there was no time to initiate a long term study of the tool, the emphasis was put towards develop the RSMLT and evaluate if it had any potential for a learning tool. The literature review revealed that visualisation is an effective method for quick information retrieval and animation as a key method to present large amounts of data as it represents movements over time and space. These aspects were reflected in the findings of the final evaluation, indicating that a tool utilizing them can become a positive contribution to a learning environment for future ATCOs.

11.2 Limitations of work

There are several limitations which should be outlined for the validity of the findings revealed in this thesis. Rogers, Sharp et al (2011) states validity is concerned about using an evaluation method that measures what is intended to measure and the actual performance of the method used. This section will describe some of the limitations of the work conducted in this thesis which may affect the validity of the findings.

As stated earlier in this thesis, the intended target group of the tool are ATCO students who can use the tool in order to train outside simulators. In this thesis, the final evaluation was conducted with students from other fields than ATC and thus may have influenced the results somewhat due to previous experiences which may have influenced ATCO students perception of this tool.

While the results from the final evaluation gave indications towars a higher level of understanding of separation rules, the usability test simulation lasted only for a fairly limited time of 20 minutes. In order to properly measure learning, one should observe participants performance over time in a longitudinal study. Measurements done in a longitudinal study should ideally reflect the users increased understanding and cababilities to apply knowledge gained through using the tool. Simulations also needs to be different from each other to avoid situations where the user is just memorizing one particular scenario rather than being able to practice aquired skills in unfamiliar scenarios. Lastly for an education tool to be successful, the cost/benefit ratio between time spent using the tool and actual learning that is being provided in the tool, leading to the question of how effective the tool is at training compared to existing methods.

12 Conclusion

The objective of the thesis was to explore the possibilities of introducing a learning tool, enabling ATCO students to practice outside of simulators and without the need of present instructors. To reach this objective, a tool for managing aircraft separation from the runway was developed in order to evaluate aspects of visualisation towards the goal of learning.

In this chapter, the research question is revisited together with a brief summary of the thesis and the findings. Lastly, proposals towards future development and research are included.

12.1 Conclusion

The research question for this thesis was the following:

What are the key factors of design when developing a prototype for the purposes of education and training of air traffic management, and what are the key visualisation techniques that should be utilized in a learning tool in order to support and create a learning experience reflecting the dynamic situation found in air traffic management?

To answer this question, the thesis have presented many important design implications when it comes to designing and developing a tool suitable for ATCO training, and for future efforts towards designing an actual system for next generation ATM systems. In order to do this, a literature review was conducted to find previous research on topics that was deemed relevant for this thesis. The topics included earlier research on ATM and the SESAR project, animation and visualisation for educational purposes, cognitive complexity as well as automation. At the early stages of this project, it was made clear that SINTEF have developed an algorithm (Kjenstad et al., 2013) which have been verified to be an improvement on several areas of ATM. An algorithm like this being implemented into future ATM systems could either be used as a decision aid tool the ATCOs could consult for near-optimal solutions or it could be part of an automated process. Research on automation was done in order to assess possible effects which different levels of automation may cause.

Reviewed work revealed several guidelines towards using visualisation and designing for learnability as well as revealing that the most prominent limiting factor towards ATC efficiency was the cognitive capabilities of human operators. Cognitive complexity and cognitive capabilities are both important aspects that should be considered when designing visualisation tools for use within ATC. To address the issue of the limited cognitive capabilities of human operators, two possibilities were identified, but not mutually exclusive;

(1) presenting visual information in such a way that does not require deep mental processing.
(2) Introducing automation for certain aspects of ATM, giving the operator fewer tasks to think about.

The gathered knowledge and the scope of the thesis resulted in a tool being developed for the purpose of testing how visualisation and animation may help students understand and learn *separation*, an aspect of ATM. The tool was evaluated through a usability test with students being participants with no prior experience with ATM in order to answer the research question. After the participants had played through the simulation, there was a debriefing session where they would fill out a short questionnaire about their experience with the tool as well as being encouraged to provide suggestions for how the tool could be improved.

The results from the final evaluation showed that there was a big difference between the participants, but still indicated that the participants gained some understanding towards the concept of separation. In a dynamic setting such as in ATC, a lot of things might happen and being in need of attention at the same time. The results from observing the usability testing sessions and the feedback from the participants indicate that the non-textual elements were the most effective in using the tool as they only needed to be looked at for a short time to deliver the information the participants were looking for. In contrast, the textual elements of the interface were understood as perfectly understandable, but it required more time and effort from the participant to retrieve information. In a hectic situation, this might be taking too much focus away from other important events that might occur at the same time, although colour usage for attention management seemed to mitigate most of these occurrences. During the debriefing, the participants expressed that they thought the tool was easy to use and that the different elements in the tool were easy to use.

12.2 Future work

This section describes some ideas for future work on developing a tool for educating ATCOs for the next generation of ATM systems. Firstly, an analytic post-simulation statistics screen is proposed to increase the efficiency of learning. Secondly, support for touch-screen devices is suggested to make the tool truly independent of location. Lastly, a proposition for the tool to include multiple scenarios is being made to ensure that understanding and knowledge gained can be applied by the student in different situations.

12.2.1 Analytic post-simulation statistics

In order for effective learning to take place it needed to evaluate and measure the ATCO students' progress in training. This could be done by ending each session with the RSMLT telling the user how well he or she did for that particular session. Statistics displayed after each session could contain a percentage on how well the student did depending on what the optimal solution would be, and also display which types of errors that occurred, the severity of them and at which frequency they occurred. By logging all of these statistics over time, a student should be able to identify which issues that require more attention in order to become more efficient at managing air traffic. Congregating data from all of the students performances throughout their learning period allows for instructors to recognize common mistakes and identify patterns that students show while using the tool. This data could be used by the instructors to know which part of the education is lacking and thus letting them know where training should be emphasized. Below is a simple suggestion of what a post-simulation statistics screen could look like in order for the students, hopefully making them aware of which mistakes they made over the duration of the simulation.

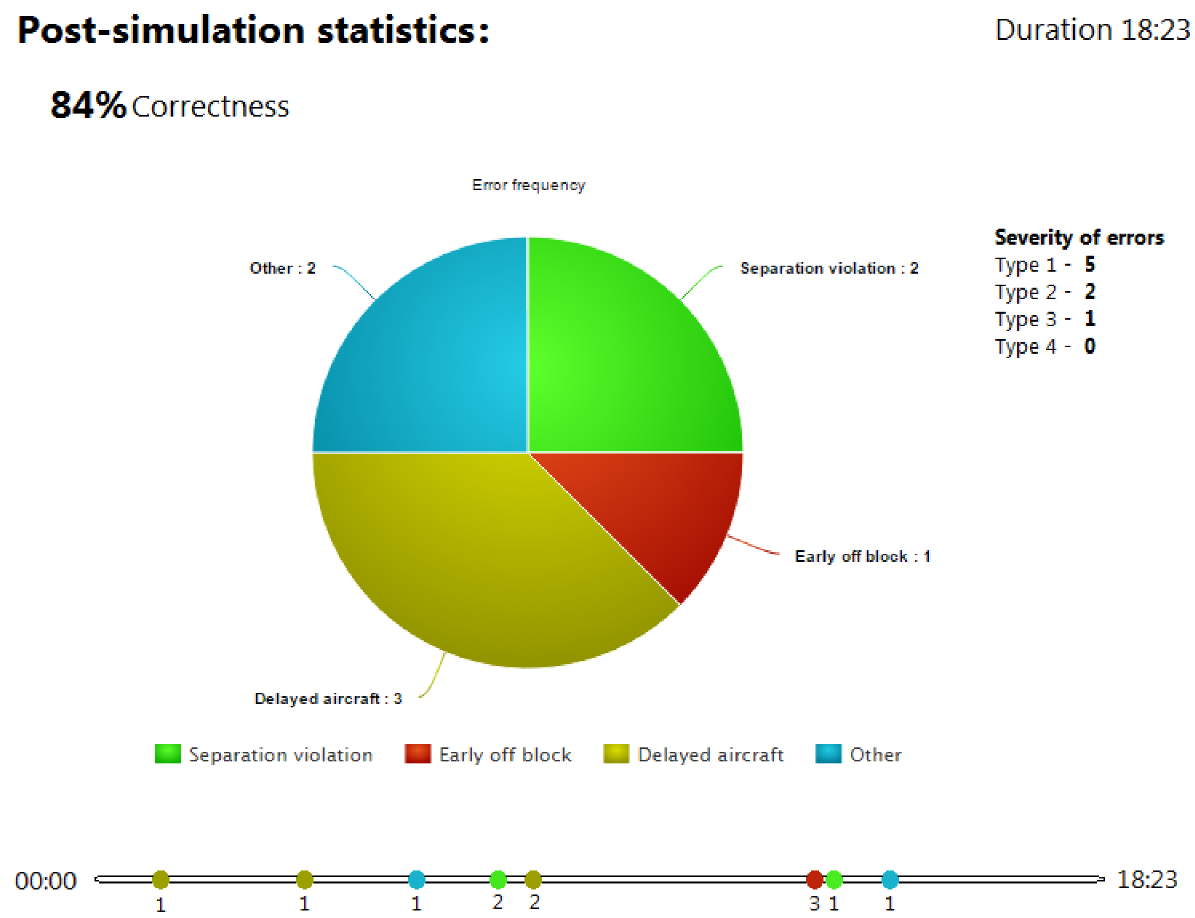


Figure 38: A suggestion for post-simulation statistics. Pie chart created on meta-chart.com.

12.2.2 Touch friendly UI

As more and more devices uses touch screen technology, efforts should be placed towards researching possibilities for ATM solutions that supports this type of technology. A learning tool supporting touch means that it could be used on devices such as tablets instead of fully fledged computers making it truly portable and supporting the notion of ATCO training being available anywhere at any time.

12.2.3 Multiple scenarios

A complete learning tool covering most, if not all aspects of ATM could benefit from being developed by using a game engine such as unity where a game consists of one or more scenes. Multiple separate scenes in the same game (tool) makes it fairly easy to implement multiple airports and multiple scenarios for each airport. The scenes could be specialized to mostly focus on few aspects of ATM or they could be defined as to represent different busy and non-busy hours based on real flight patterns. Another idea is taken from the theory of using gamification to increase user engagement in non-gaming contexts by rewarding the user (Deterding et al., 2011). Initially, the tool could present the user with one scenario available which requires the user to complete it in order to unlock the next and more advanced one. This could give the user some sense of progress and achievement while making the tool and the process of learning a more exciting experience. On scenario completion, the user could be presented with a three star rating based on how well he or she performed compared to the optimal solution. 75 - 89 % could be one star, 90 - 98% two stars and 99 - 100% three stars. The stars may entice users to revisit previously “won” scenarios in order to improve their previous result, further expanding their understanding of ATM.

13 References

AINSWORTH, S. 2006. DeFT: A conceptual framework for considering learning with multiple representations. *Learning and Instruction,* 16**,** 183-198.

AIRSERVICES AUSTRALIA. 2015. *Separation standards* [Online]. www.airserviceaustralia.com. Available: www.airservicesaustralia.com/services/how-air-traffic-control-works/separation-standards/ [Accessed 19/05 2016].

AVINOR. n.d. Available: https://avinor.no/bli-flygeleder/opptak/ [Accessed 22/01 2016].

BAGOZZI, R. P., DAVIS, F. D. & WARSHAW, P. R. 1989. User Acceptance of Computer Technology: A Comparision of Two Theoretical Models. *Management Science,* 35**,** 982-1003.

BALDONADO, M. Q. W., KUCHINSKY, A. & WOODRUFF, A. 2000. Guidelines for Using Mutliple Views in Information Visualization.

BARR, P., NOBLE, J. & BIDDLE, R. 2003. Icons R Icons. *Australasian User Interface Conference,* 18.

BEAR, J. H. 2015. *Color Symbolism: What Different Colors Mean to Us* [Online]. Available: http://desktoppub.about.com/cs/color/a/symbolism.htm [Accessed 21/01 2016].

BEDFORD, A. 2014. *Icon Usability* [Online]. Available: https://www.nngroup.com/articles/icon-usability/ [Accessed 21/01 2016].

BENSON, J. 2014. *The Birth, Life and Certification of a Very Good Idea* [Online]. Smidig 2014. Available: http://vimeo.com/album/3107510/video/110550724 [Accessed March 27nd 2016].

BESSENSEN, M. S., DAHLE, T., JOSEPHSEN, A. & PETTERSEN-HJELVIK, M. 2014. Flashback post simulation tool.

BOUWER, J., MAXWELL, D. & SAXON, S. 2015. *Gridlock on the ground: How airlines can respond to airport congestion* [Online]. McKinsey&Company. Available: www.mckinsey.com/industries/travel-transport-and-logistics/our-insights/gridlock-on-the-ground-how-airlines-can-respond-to-airport-congestion [Accessed 10/03 2016].

BRANDT, E. 2007. How tangible mock-ups support design collaboration. *Knowledge, Technology & Policy,* 20**,** 179-192.

BRATTETEIG, T. 2004. *Making change, Dealing with relations between design and use.* PhD, University of Oslo.

CHAPMAN, C. 2010. *Color Theory for Designers, Part 1: The Meaning of Color* [Online]. Available: https://www.smashingmagazine.com/2010/01/color-theory-for-designers-part-1-the-meaning-of-color/ [Accessed 21/01 2016].

CILLIERS, P. 1998. *Complexity and Postmodernism: Understanding Complex Systems*, Routledge.

COLOR WHEEL PRO. n.d. *Color Meaning* [Online]. Available: http://www.color-wheel-pro.com/color-meaning.html [Accessed 21/01 2016].

COLORMATTERS. n.d. *Basic Color Theory* [Online]. Available: http://www.colormatters.com/color-and-design/basic-color-theory [Accessed 22/01 2016].

CORBIN, J. & STRAUSS, A. 1990. Grounded Theory Research: Procedures, Canons, and Evaluative Criteria. *Qualitaive Sociology,* 13**,** 1-19.

DEPARTMENT OF HEALTH AND HUMAN SERVICES DEPT (U.S.) 2006. Usability Testing. *Research-Based Web Design & Usability Guidelines.* Author.

DERBER, A. 2010. *Europe and USA take step towards ATM interoperability* [Online]. Flightglobal. Available: www.flightglobal.com/news/articles/europe-and-usa-take-step-towards-atm-interoperabilit-343582/ [Accessed 03/04 2016].

DETERDING, S., SICART, M., NACKE, L., O'HARA, K. & DIXON, D. Gamification: Using Game Design Elements in Non-Gaming Contexts. CHI, 2011 Vancouver, BC, Canada.
DOLAN, A. 2015. *The anatomy of a flight strip* [Online]. Available: http://nats.aero/blog/2015/05/the-anatomy-of-a-flight-strip/ [Accessed February 7 2016].
EASON, K. 1988. *Information Technology and Organisational Change,* London, Taylor and Francis.
EISENSTEIN, P. A. 2014. *Cleaning Up as Car Wash Industry Turns 100* [Online]. Available: http://www.thedetroitbureau.com/2014/04/cleaning-up-as-car-wash-industry-turns-100/ [Accessed 24 Feb 2015].
ENDSLEY, M. R. & JONES, D. G. 2012. *Designing for Situation Awareness, An Approach to User-Centered Design*, CRC Press Taylor & Francis Group.
EUROCONTROL. *What is air traffic management?* [Online]. Available: www.eurocontrol.int/articles/what-air-traffic-management [Accessed February 1 2016].
EUROCONTROL. 2015. *Forecasts* [Online]. Eurocontrol. Available: www.eurocontrol.int/articles/forecasts [Accessed 10/03 2016].
FAA 2015. FAA Forecast Sees Continued, Steady Growth in Air Travel.
FATLAND, O. G., FURUBERG, E., VAN DIJK, L. W. & DÆHLEN, Å. K. 2014. Goodgate, Realtime air traffic management tool.
FEKETE, G. 2008. *Designing The Holy Search Box: Examples And Best Practices* [Online]. Available: www.smashingmagazine.com/2008/12/designing-the-holy-search-box-examples-and-best-practices/ [Accessed February 5th 2016].
FLOYD, C. 1984. A systematic look at prototyping. *In:* REINHARD, B., KUHLENKAMP, K., MATHIASSEN, L. & ZÜLLIGHOVEN, H. (eds.) *Approaches to Prototyping.* New York: Springer-Verlag.
GREENOUGH, J. 2016. *10 million self-driving cars will be on the road by 2020* [Online]. Available: http://www.businessinsider.com/report-10-million-self-driving-cars-will-be-on-the-road-by-2020-2015-5-6?r=US&IR=T [Accessed 19/06 2016].
HELGASON, D. 2012. Game developers, start your Unity 3D engines. *In:* TAKAHASHI, D. (ed.).
HEWES, R. W. 1951. *Hewes Flight Progress Board*.
HILBURN, B. 2004. Cognitive Complexity in Air Traffic Control: A Literature Review. Center for Human Performace Research.
HISTON, J. M., AIGOIN, G., DELAHAYE, D., HANSMAN, J. R. & PUECHMOREL, S. Introducing Structural Considerations into Complexity Metrics. USA / Europe Air Traffic Management R&D Seminar, 2001 Santa Fe.
HISTON, J. M. & HANSMAN, J. R. 2008. *Mitigating Complexity in Air Traffic Control: The Role of Structure-Based Abstractions.* PhD, MIT International Center for Air Transportation (ICAT). Massachusetts Institute of Technology, Cambridge, USA.
HOUDE, S. & HILL, C. 1997. What do Prototypes Prototype? *In:* HELANDER, M. G., LANDAUER, T. K. & PRABHU, P. V. (eds.) *Handbook of Computer Interaction.* 2 ed. Amsterdam: Elsevier Science.
IATA 2015. IATA Air Passenger Forecast Shows Dip in Long-Term Demand. The International Air Transport Association.
ISAIAS, P. & ISSA, T. 2015. Information System Development Life Cycle Models. *High Level Models and Methodologies for Information Systems.* 1 ed. New York: Springer-Verlag.
IVAO. 2016. *Holding Stack Management* [Online]. Available: www.ivao.aero/training/documentation/index.php?section=apc [Accessed 29/03 2016].
JEFFRIES, R. & MILLER, J. R. 1991. User Interface Evaluation in the Real World: A Comparison of Four Techniques. *HPL.* Hewlett Packard Labs.
KAASBØLL, J. 2016. *Developing digital competence - learning, teaching and supporting use of information technology*, Unpublished.
KIM, J. H. & LEE, K. P. 2005. Cultural Difference and Mobile Phone Interface Design: Icon Recognition According to Level of Abstraction. *Proceedings from MobileHCI'05*.
KJENSTAD, D., MANNINO, C., SCHITTEKAT, P. & SMEDSRUD, M. Integrated Surface and Departure Management at Airports by Optimization. Modeling, Simulation and Applied Optimization (ICMSAO), 2013 5th International Conference on, 2013 Hammamet. IEEE, 1-5.

KOUTSOURELAKIS, C. & CHORIANOPOULOS, K. 2010. Icons in mobile phones, Comprehensibility differences between older and younger users. *Information Design Journal,* 18**,** 22-35.

LANKTON, P. 2007. *Endsley's model* [Online]. Available: http://en.wikipedia.org/wiki/User:Dr._Peter_Lankton [Accessed 25 Feb 2015].

LAZAR, J., FENG, J. H. & HOCHHEISER, H. 2010. *Research Methods In Human Computer Interaction*, John Wiley and Sons.

LEVKOWITZ, H. 1997. *Color Theory and Modeling for Computer Graphics, Visualization, and Multimedia Applications,* University of Massachusetts Lowell, Lowell, Massachusetts, USA, Kluwer Academic Publishers.

LEWIS, C., POLSON, P., RIEMAN, J. & WHARTON, C. 1999. The Cognitive Walkthrough Method: A Practitioner's Guide. *In:* NIELSEN, J. M., ROBERT L (ed.) *Usability Inspection Methods.* Wiley.

LOWE, R. K. 2001. Beyond "eye candy": improving learning with animations.

LOWE, R. K. 2003. Animation and learning: selective processing of information in dynamic graphics. *Learning and Instruction***,** 157-176.

LÖWGREN, J. & STOLTERMAN, E. 2004. *Thoughtful Interaction Design: A Design Perspective on Information Technology*, Cambridge MIT Press.

LOWY, J. 2016. *No room for old technology in new airport towers* [Online]. Associated Press. Available: http://www.apnewsarchive.com/2016/Union%3A_No_room_for_old_technology_in_new_airport_towers/id-7b29fadaf32c441f84d88dd7d90cfbb7 [Accessed 16/06 2016].

MADHAVAN, P. & WIEGMANN, D. A. 2004. A New Look at the Dynamics of Human-Automation Trust. *PROCEEDINGS of the HUMAN FACTORS AND ERGONOMICS SOCIETY 48th ANNUAL MEETING*.

MAJUMDAR, A. & POLAK, J. W. 2001. Estimating Capacity of Europe's Airspace Using a Simulation model of Air Traffic Controller Workload. *Transportation Research Journal of the Transportation Resarch Board*.

MAYER, R. E. & ANDERSON, R. B. 1992. The Instructive Animation: Helping Students Build Connections Between Words and Pictures in Multimedia Learning. *Journal of Educational Psychology,* 84**,** 444-452.

MAYER, R. E. & MORENO, R. 2002. Animation as an Aid to Multimedia Learning. *Educational Psychology Review,* 14**,** 87-99.

MÖLLER, T. & TORY, M. 2005. Evaluating Visualizations: Do Expert Reviews Work? *IEEE Computer Graphics and Applications,* 25**,** 8-11.

MORAY, N., INAGAKI, T. & ITOH, M. 2000. Adaptive Automation, Trust, and Self-Confidence in Fault Management of Time-Critical Tasks. *Journal of Experimental Psychology: Applied,* 6**,** 44-58.

MOWAT, J. 2002. Cognitive Walkthroughs: Where they came from, what they have become, and their application to EPSS design. The Herridge Group Inc.

NATS. 2015. *The anatomy of a control tower* [Online]. Available: http://nats.aero/blog/infographic-the-anatomy-of-a-control-tower/ [Accessed February 7 2016].

NIELSEN, J. 1995a. *How to Conduct a Heuristic Evaluation* [Online]. Available: http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/ [Accessed 18 Sept 2015].

NIELSEN, J. 1995b. *Icon Usability for the 1995 Sun Microsystems' Website* [Online]. Available: http://www.nngroup.com/articles/icon-usability-1995-sun-microsystems-website/.

NIELSEN, J. 1995c. *Severity Ratings for Usability Problems* [Online]. Available: https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/ [Accessed 02/10 2015].

ØDEGÅRD, S. S. 2013. *Exploring Visualisation - Solutions for Air Traffic Control Workflow Productivity Improvement.* Master, University of Oslo.

PENZO, M. 2006. *Evauluating the Usability of Search Forms Using Eyetracking: A Practical Approach* [Online]. Available: www.uxmatters.com/mt/archives/2006/01/evaluating-the-usability-of-search-forms-using-eyeytracking-a-practical-approach.php [Accessed February 5 2016].

RIEBER, L. P. 1989. Animation in Computer-Based Instruction. *Educational Technology Research and Development,* 38**,** 77-86.
ROGERS, Y., SHARP, H. & PREECE, J. 2011. *Interaction Design, beyond human-computer interaction*, Wiley.
SCAIFE, M. & ROGERS, Y. 1996. External cognition: how do graphical representations work. *Human-Computer Studies,* 45**,** 185-213.
SCHNEIDERMAN, B. & PLAISANT, C. Strategies for Evaluating Information Visualization Tools: Multi-dimensional In-depth Long-term Case Studies. Advanced Visual Interfaces Conference, 2006 Venice. 1-7.
SCHWAB, S. 2012. *VFR VS. IFR Flying* [Online]. www.stephan-schwab.com. Available: www.stephan-schwab.com/airtrave/vfr-ifr.html [Accessed 19/05 2016].
SCOTLAND, K. 2010. Aspects of Kanban. *Methods & Tools,* 19**,** 3-14.
SKYBRARY. n.d. *Separation Standards* [Online]. www.skybrary.aero. Available: www.skybrary.aero/index.php/Separation_Standards [Accessed 19/05 2016].
SMITH, M. J. & SALVENDY, G. 2001. Cross-Cultural User-Interface Design. *Proceedings from the Human-Computer Interface Internat (HCII) 2001,* 2**,** 502-505.
SOMMERVILLE, I. 2011. *Software Engineering,* Boston, Pearson.
SPOOL, J. 2006. *Two Simple Post-Test Questions* [Online]. Available: https://www.uie.com/brainsparks/2006/03/23/two-simple-post-test-questions/ [Accessed 21/07 2016].
SUCHMAN, L. 2002. Located accountabilities in technology production. *Scandinavian Journal of Information Systems,* 14**,** 91-105.
TIGERCOLOR. n.d. *How many colors should you use in your designs?* [Online]. Available: http://www.tigercolor.com/color-lab/tips/tip-01.html [Accessed 22/01 2016].
TSAI, P. 1996. A survey of empirical usability evaluation methods. GSLIS Indep. study.
TVERSKY, B. & MORRISON, J. B. 2001. Animation: can it facilitate? *Int. J. Human-Computer Studies***,** 247-262.
UNITY. 2015a. *Build Once Deploy Anywhere - Industry-leading Multiplatform Support* [Online]. Available: www.unity3d.com/unity/multiplatform [Accessed 13/10 2015].
UNITY. 2015b. *The Leading Global Game Industry Software* [Online]. Available: www.unity3d.com/public-relations [Accessed 13/10 2015].

Appendix A – Consent form, usability testing

Samtykkeskjema om deltagelse i brukbarhetstesting av prototype om visualisering og læring.

Ansvarlig
Lars Ivar Kleven larsikl@student.matnat.uio.no

Bakgrunn og formål
Gjennom min masteroppgave har jeg utviklet en prototype som benytter seg av ulike visualiseringsløsninger for å fremme læring hos flygelederstudenter. Masteroppgaven skrives I samarbeid med SINTEF og er en del av et større internasjonalt prosjekt med navnet SESAR. SESAR prosjektet sikter på å overhale de eksisterende flytrafikksystemene I Europa.

Formålet med denne studien er å teste en prototype gjennom brukbarhetstesting. Dette gjøres for å få ett inblikk i hvordan brukere forholder seg til og bruker prototypen.

Hva innebærer deltagelse I studien?
Å delta i studien vil ta ca 30 minutter, hvorav de første 20 minuttene vil bli brukt på å interagere med prototypen. De siste 10 minuttene er satt av til å diskutere opplevelsen rundt bruken av prototypen.

Hva skjer med informasjonen om deg?
Alle personopplysninger vil bli behandled konfidensielt. Det er kun ansvarlig som vil ha tilgang til eventuelle personopplysninger.

Frivillig deltakelse
Deltagelse i studien er frivillig, og du kan når som helst trekke deg fra studien uten å oppgi noen grunn. Dersom du skulle trekke deg vil fortsatt alle eventuelle personopplysninger være anonymisert.

Samtykke til deltagelse i studien

Ved å skrive under på dette skjemaet sier du det villig til å delta i studien og at du har satt deg inn i informasjonen gitt ovenfor.

Signatur ____

Sted og Dato ____

Appendix B.1 - Results from search bar experiment, p1

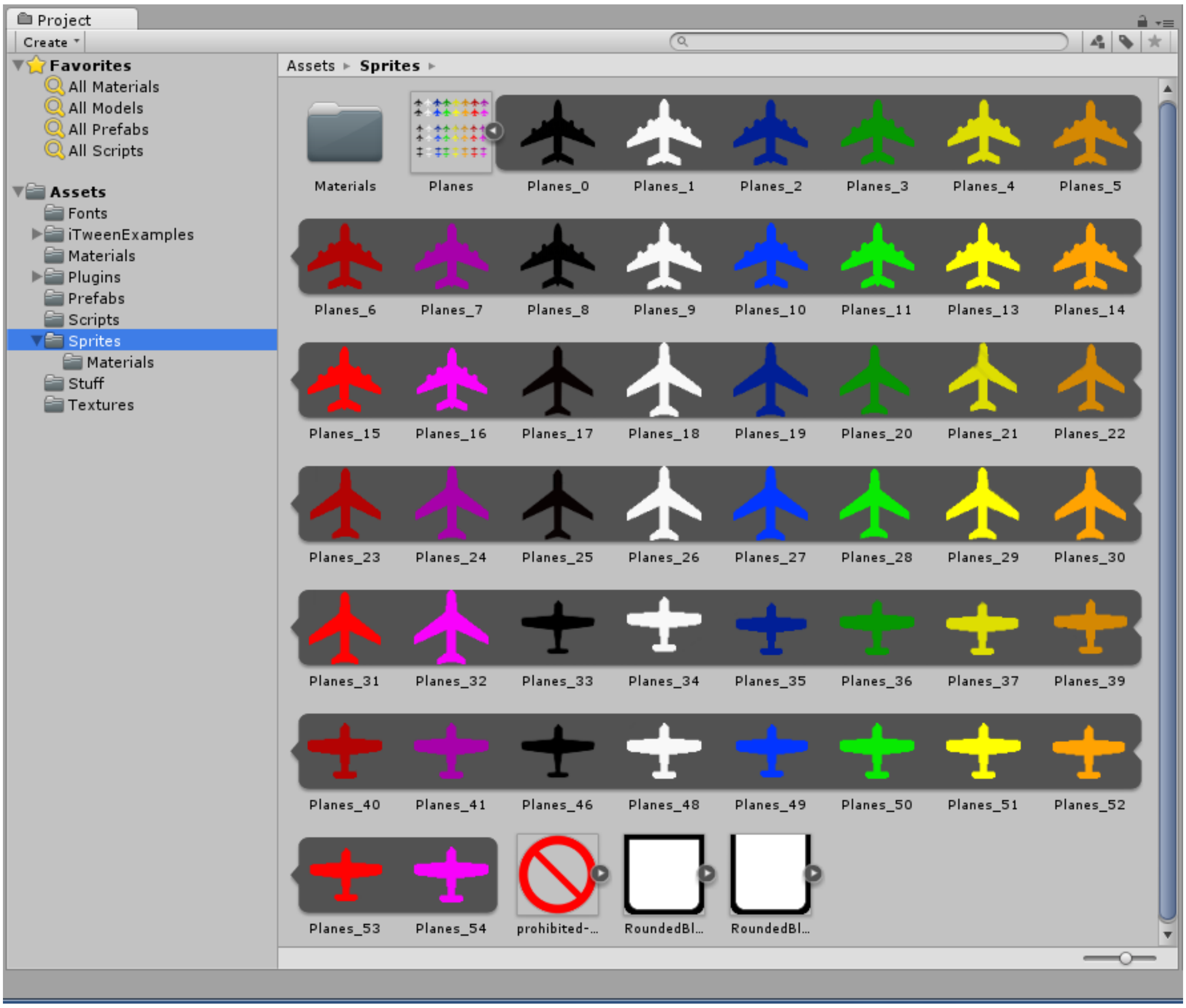
	This research was undertaken February 5th 2016. Any website may have changed since then.			
#	website	search box placement	symbol/button	notes
1	www.wikipedia.no	Centre	"search" button in norwegian, new-norwegian and sami.	When on an article page, the search box moves to the upper right corner. Magnifying glass.
2	www.thefreedictionary.com	Upper centre	Right arrow	
3	www.google.no	Centre	"google search"	button below search box
4	Google results page	Upper left	Magnifying glass	
5	www.thesaurus.com	Upper centre	Magnifying glass	
6	www.econsultancy.com	Upper right	Magnifying glass	
7	ux.stacexchange.com	Upper right	Magnifying glass	
8	www.komplett.no	Upper centre	Magnifying glass	
9	www.ps.no	Upper left	"search" button	
10	www.cdon.no	Upper centre	"search" button	
11	www.lekmer.no	Upper centre	Magnifying glass	
12	www.rakuten.co.uk	Upper centre	Magnifying glass	
13	www.spillsjefen.no	Upper centre	Magnifying glass	
14	www.facebook.com	Upper left	Magnifying glass	
15	www.smashingmagazine.com	Upper right	"search" button	
16	www.uxmatters.com	Upper right	"search" button below	
17	www.telenor.no	Upper right	Magnifying glass	
18	www.vg.no	Upper right	Magnifying glass, no search box	when pressed makes search box appear over other content
19	www.dagbladet.no	Upper right	Magnifying glass	
20	www.amazon.com	Upper centre	Magnifying glass	

Appendix B.2 - Results from search bar experiment, p2

21	www.ebay.com	Upper centre	"search" button	
22	www.assetstore.unity3d.com	Upper centre	Magnifying glass	
23	forum.worldoftanks.eu	Upper right	Magnifying glass	
24	www.youtube.com	Upper centre	Magnifying glass	
25	www.tine.no	Upper right	Magnifying glass	
26	www.dnb.no	Upper centre	Magnifying glass	
27	aurskog-sparebank.no	Upper right	Magnifying glass	
28	plus.google.com	Upper centre	Magnifying glass	
29	www.twitch.tv	Upper left	Magnifying glass	
30	www.adobe.com	Upper right	Magnifying glass, no search box	when pressed makes a huge search box covering most of the screen
31	www.hotels.com	Upper left	Crosshair	
32	www.trivago.no	Centre	"search" button	
33	www.norwegian.no	Centre	"search & order" button	
34	www.sas.no	Upper right	Magnifying glass.	This site also got another search area for flights with a "search" button
35	www.bt.no	Hidden (upper left)	Magnifying glass	Menu button that opens a menu bar containing a search box
36	www.time.com	Upper left	Magnifying glass, no search box	When pressed makes search box appear over other content
37	www.zam.com	Upper right	Magnifying glass	
38	wiki.wargaming.net	Upper right	"search" button	
39	superuser.com	Upper right	Magnifying glass	
40	no.pinterest.com	Upper centre	No button	Generates "tags" from a dropdown menu based on input in the text field

Appendix C – Sprite sheet

Sprite Sheet sliced into 36 different aircraft sprites



Appendix D – Tasks from expert-review

Description

This expert-review is a part of an ongoing master thesis in interaction design focusing on learning with visualisation, specifically within the domain of air traffic management being a part of the SESAR project.

In time, as a part of the SESAR project there is a wish to develop a complete learning tool that can be used outside the air traffic management simulators and without the presence of instructors. The result will be a far more flexible learning situation where air traffic students are enabled to learn anytime, anywhere.

The screenshots included in this set of user tasks are taken from an experimental prototype which aims at visualising the ***separation rules*** describing how much time should pass between each aircraft of different types/sizes.

The **evaluation objective** is to evaluate this prototype in order to identify usability issues.

User Task 1

<u>What to do:</u>

1. Use the screenshot labelled **Screenshot 1.1** in order to do the tasks 1.1 to 1.4 below. If the tasks indicates that an answer should be given, write down your answer on the line below the question.
2. Describe any problems found using the forms labelled **User task 1, feedback.**
3. Use the severity scale located at the bottom of the feedback forms to rate the indicated problems.

1.1. Which aircraft is currently selected?

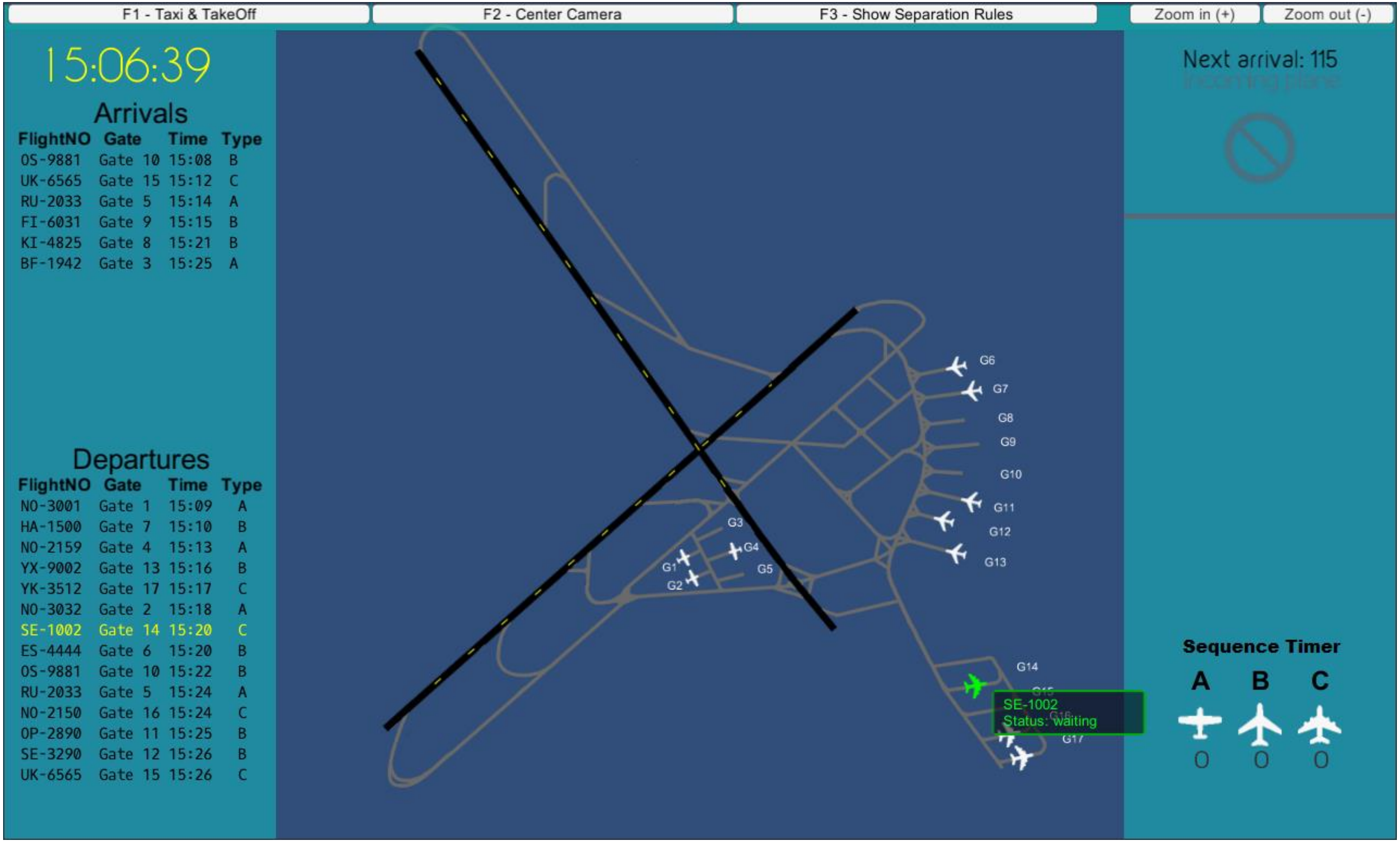
a.	Flight number:
b.	Gate number:
c.	Departure time:
d.	Aircraft type:

1.2. In how long will the next aircraft land?

1.3 Which **type** of aircraft got the flight number RU-2033?

1.4 Which aircraft(s) are using or will be using gate 15? (write down the **flight number**)

Screenshot 1.1



User task 1, feedback.   Page 1

Question#	Problem / comment	Severity of the problem*

*0 = I don’t agree that this is a usability problem at all
1 = Cosmetic problem only, does not need fixing unless time allows it
2 = Minor usability problem: low priority problem
3 = Major usability problem: important to fix, high priority
4 = Usability disaster: this must be fixed, highest priority

User task 1, feedback.   Page 2

Question#	Problem / comment	Severity of the problem*

*0 = I don’t agree that this is a usability problem at all
1 = Cosmetic problem only, does not need fixing unless time allows it
2 = Minor usability problem: low priority problem
3 = Major usability problem: important to fix, high priority
4 = Usability disaster: this must be fixed, highest priority

User Task 2

<u>What to do:</u>

1. Use the screenshots labelled **Screenshot 2.1**, **2.2** and **2.3** in order to do the tasks 2.1 to 2.4 below. If the tasks indicates that an answer should be given, write down your answer on the line below the question.
2. Describe any problems found using the forms labelled **User task 2, feedback.**
3. Use the severity scale located at the bottom of the feedback forms to rate the indicated problems.

2.1 Click to make the separation rules visible

2.2 Use the table below:
Imagine the aircraft **type** in the left column have just departed from the airport.
Figure out which of the waiting aircrafts that should be next in sequence based on the separation rules and fill in the right column labeled “Next in sequence”

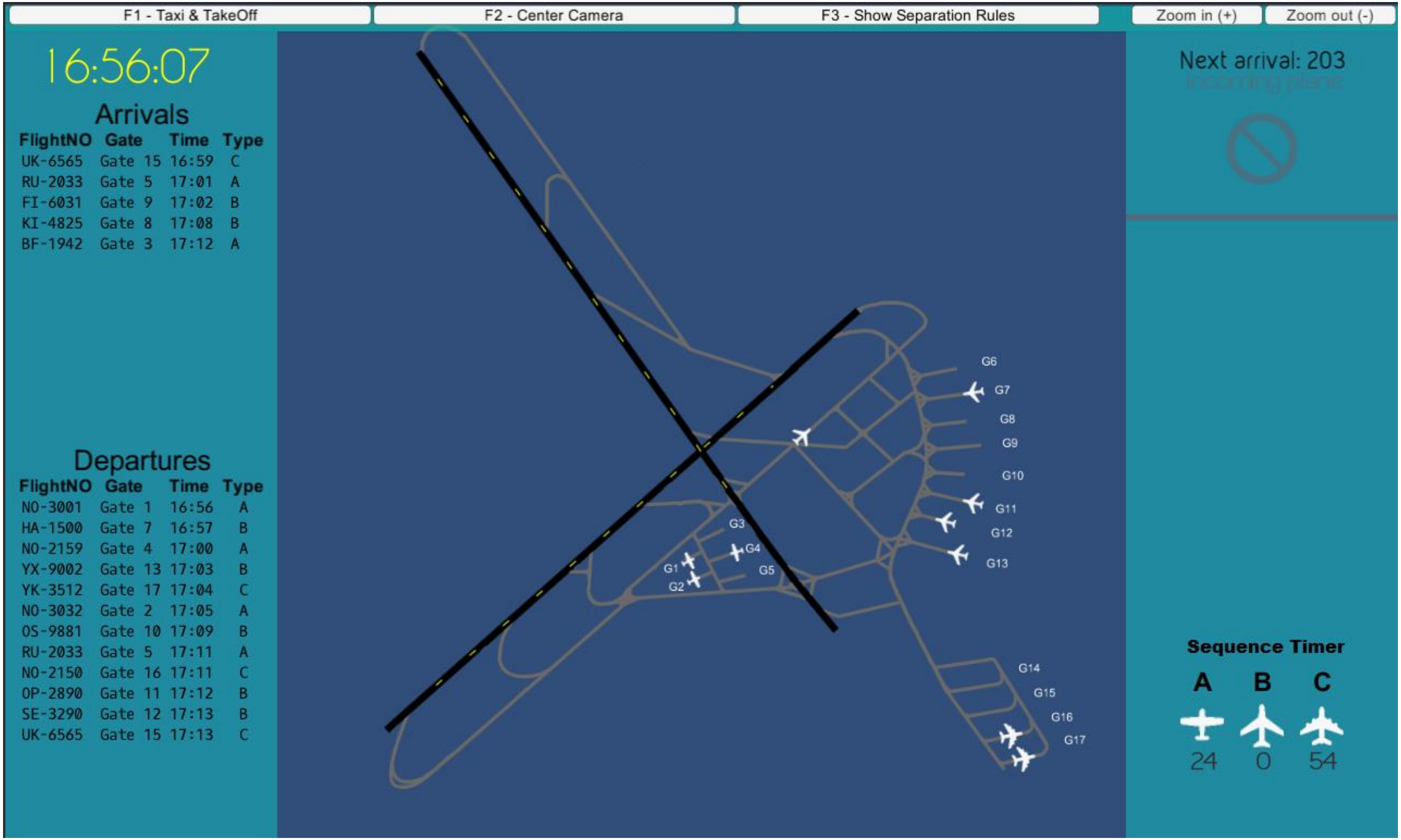
Just left the airport	Waiting aircraft	Next in sequence
C	B A	
B	A C	
A	C B	
B	A B C	

2.3 By looking at the screenshot, can you figure out which type of aircraft that most recently departed?

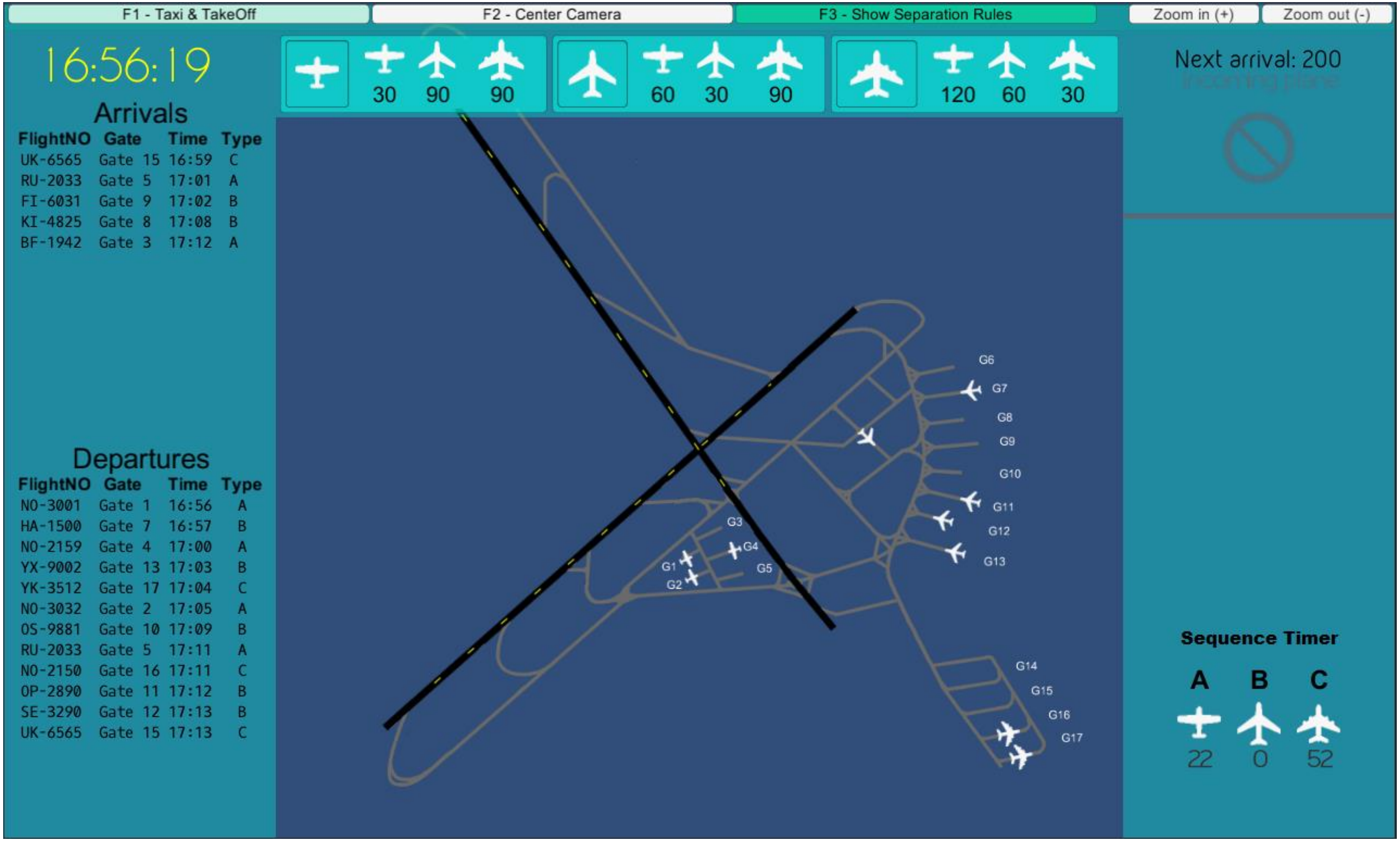
____

2.4 Click to hide the separation rules

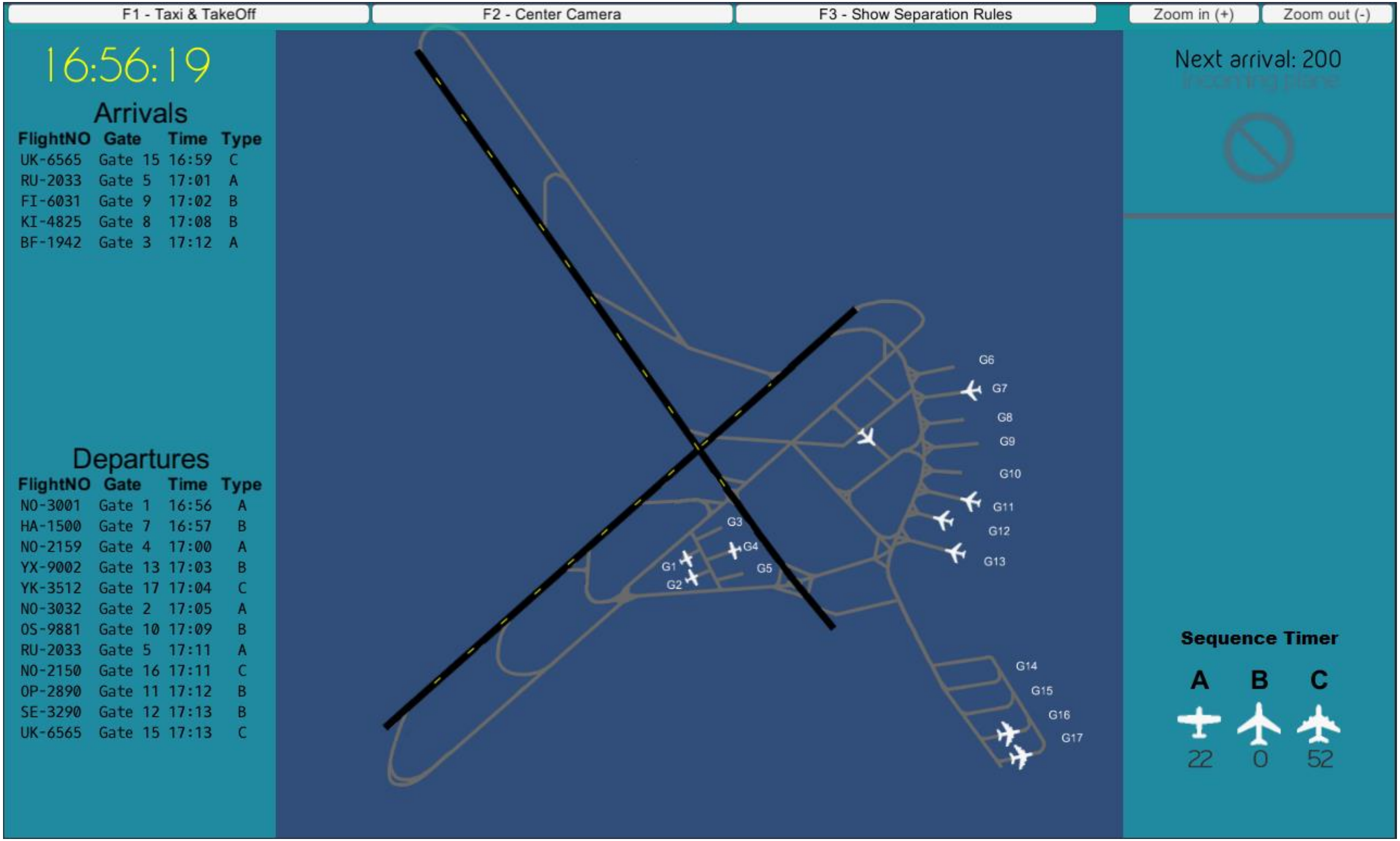
Screenshot 2.1



Screenshot 2.2



Screenshot 2.3



User task 2, feedback   Page 1

Question#	Problem / comment	Severity of the problem*

*0 = I don’t agree that this is a usability problem at all
1 = Cosmetic problem only, does not need fixing unless time allows it
2 = Minor usability problem: low priority problem
3 = Major usability problem: important to fix, high priority
4 = Usability disaster: this must be fixed, highest priority

User task 2, feedback   Page 2

Question#	Problem / comment	Severity of the problem*

*0 = I don’t agree that this is a usability problem at all
1 = Cosmetic problem only, does not need fixing unless time allows it
2 = Minor usability problem: low priority problem
3 = Major usability problem: important to fix, high priority
4 = Usability disaster: this must be fixed, highest priority

User Task 3

<u>What to do:</u>

Use the screenshots labelled Screenshot 3.1, 3.2 and 3.3 in order to do the tasks 3.1 to 3.4 below. If the tasks indicates that an answer should be given, write down your answer on the line below the question.

Describe any problems found using the forms labelled User task 3, feedback.

Use the severity scale located at the bottom of the feedback forms to rate the indicated problems

When should the next aircraft arrive?

____

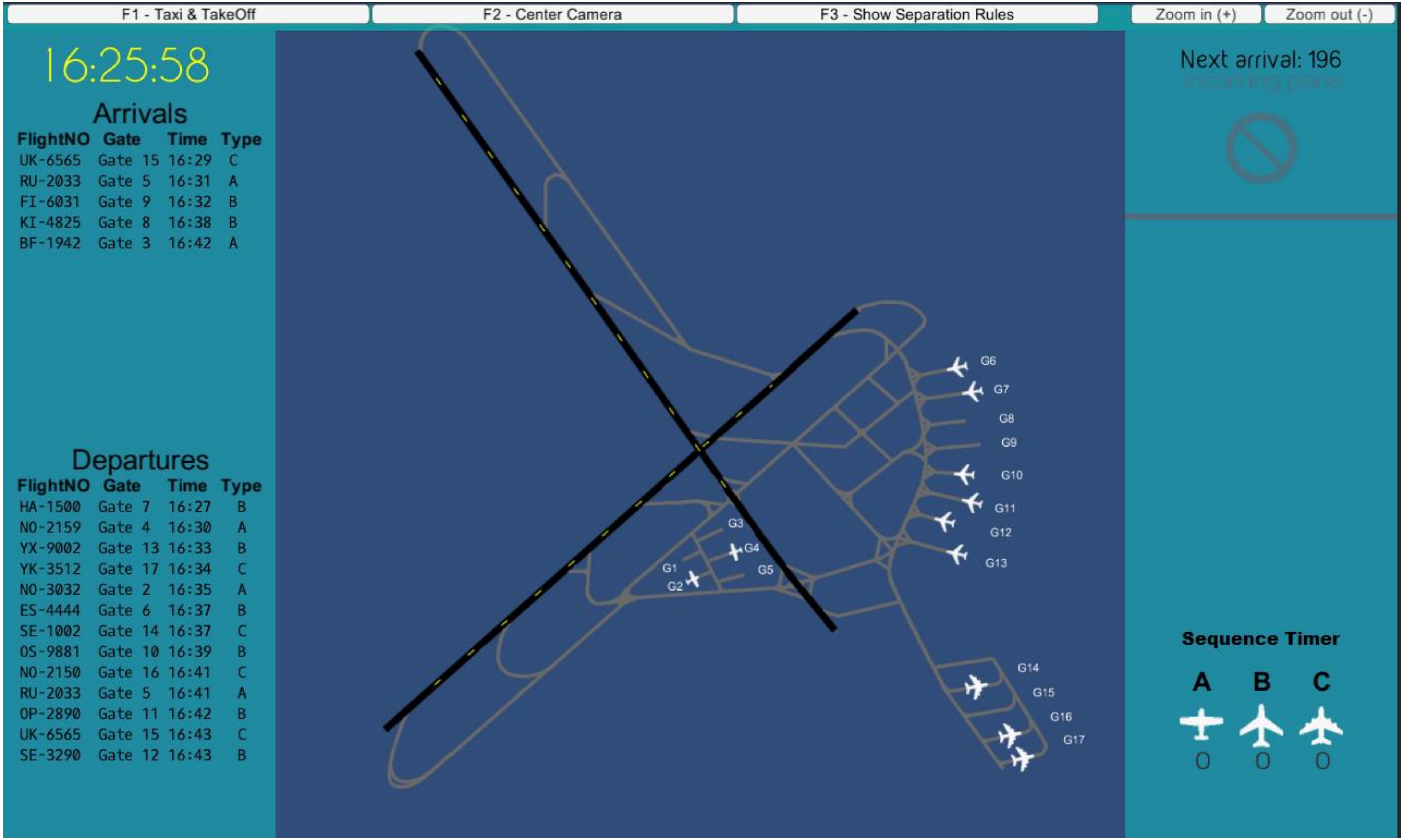
How long should you wait before giving orders for the first scheduled aircraft to depart?

____

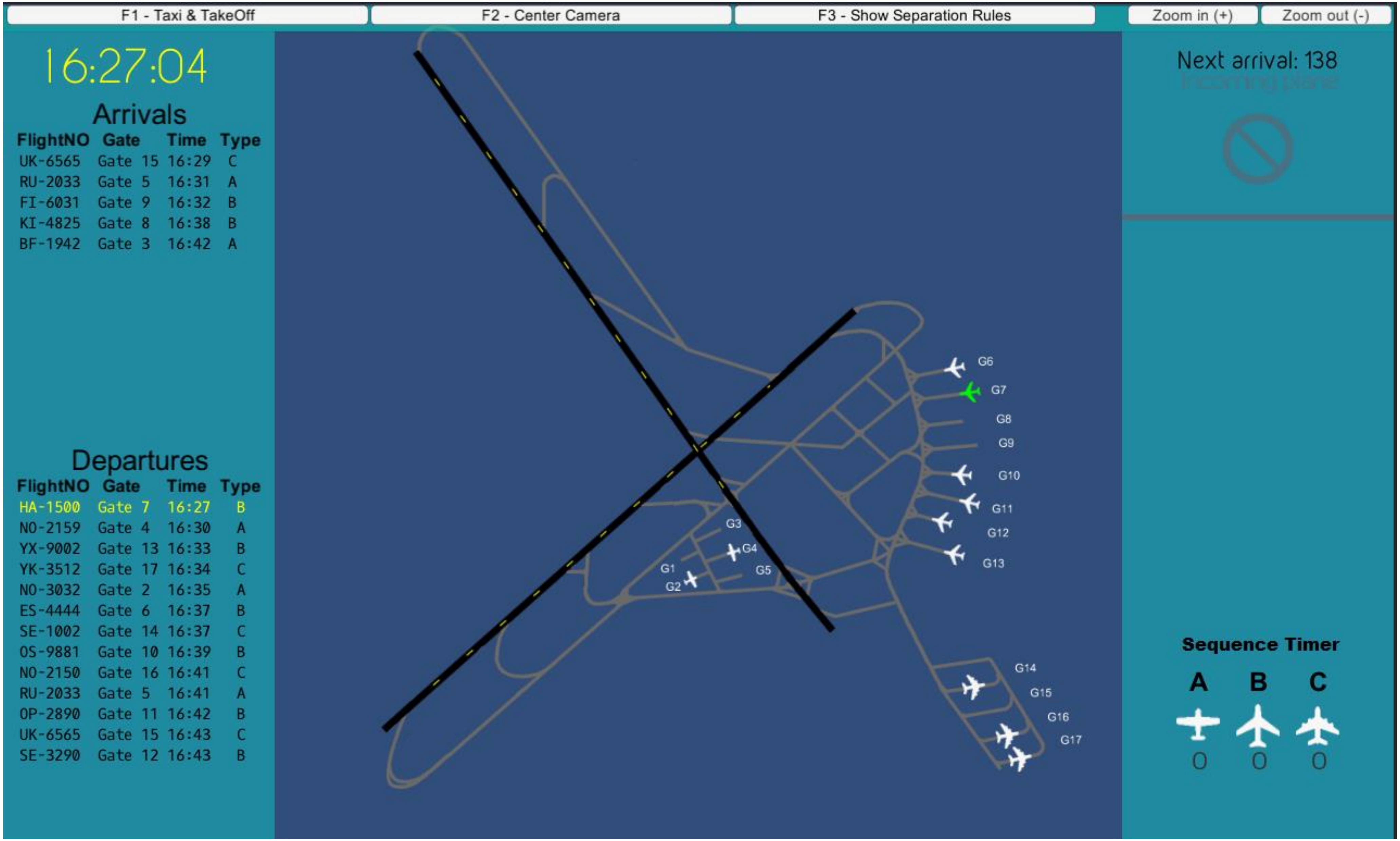
Select the aircraft scheduled to depart at 16:27

Click to send the selected aircraft to taxi and depart from the airport

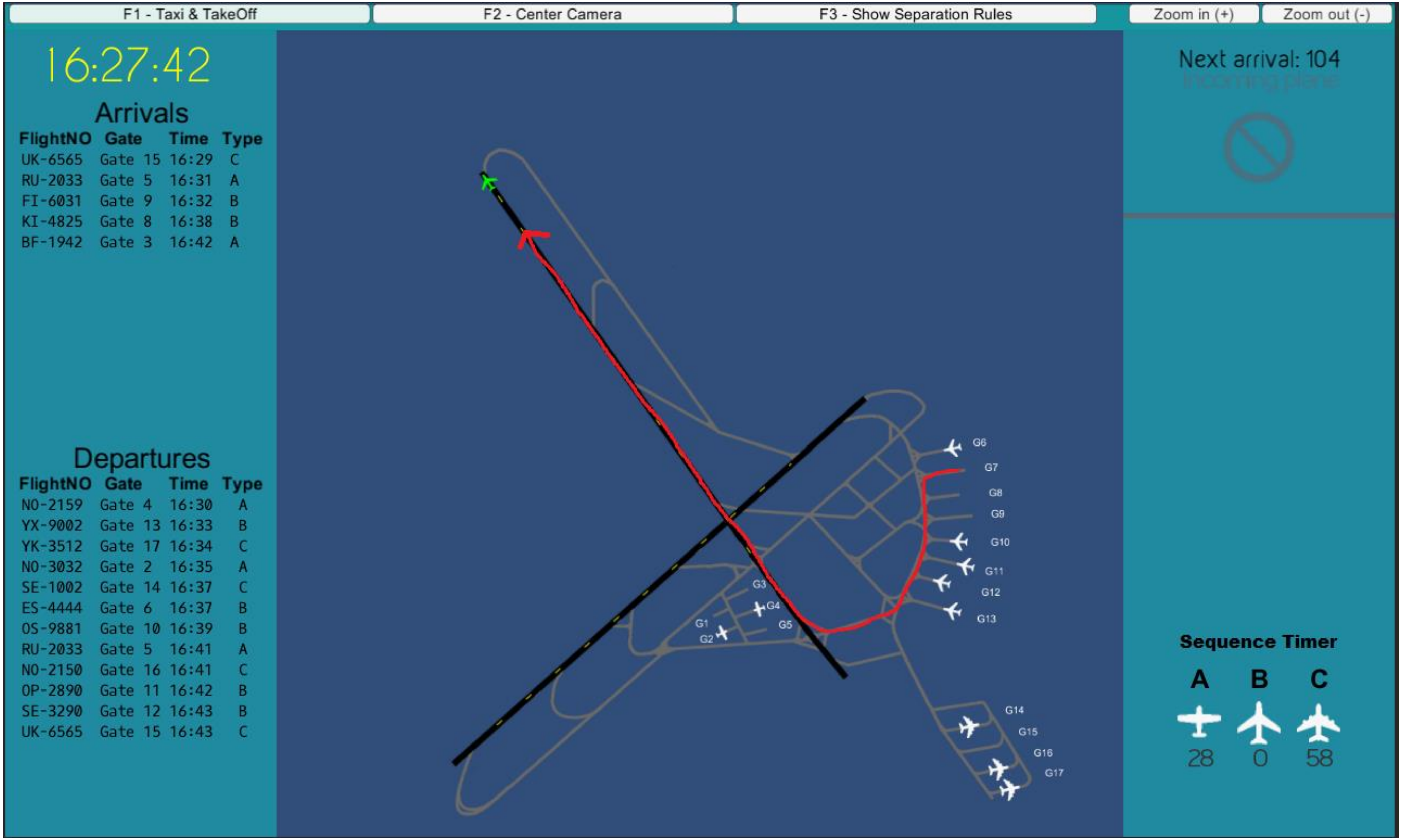
Screenshot 3.1



Screenshot 3.2



Screenshot 3.3



User task 3, feedback   Page 1

Question#	Problem / comment	Severity of the problem*

*0 = I don't agree that this is a usability problem at all
1 = Cosmetic problem only, does not need fixing unless time allows it
2 = Minor usability problem: low priority problem
3 = Major usability problem: important to fix, high priority
4 = Usability disaster: this must be fixed, highest priority

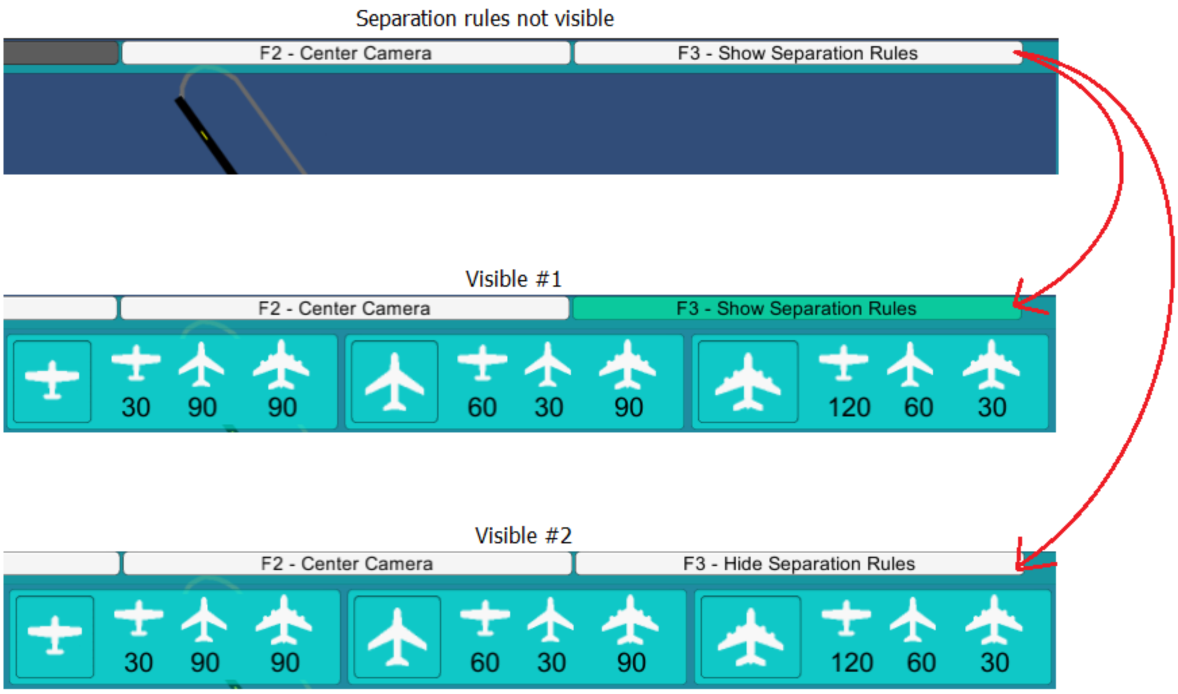
User task 3, feedback   Page 2

Question#	Problem / comment	Severity of the problem*

*0 = I don’t agree that this is a usability problem at all
1 = Cosmetic problem only, does not need fixing unless time allows it
2 = Minor usability problem: low priority problem
3 = Major usability problem: important to fix, high priority
4 = Usability disaster: this must be fixed, highest priority

Additional

Which of the following solutions are the most intuitive to you, #1 or #2? Feel free to explain why.



If you have any other comments about the prototype, the tasks or the presentation, please write the backside of any page

Thank you for your participation, it is greatly appreciated.

Appendix E – Usability testing questionnaire

Sett ring rundt det tallet som representerer din opplevelse.

Q1: Det var lett å få oversikt over komponentene i verktøyet

Sterkt uenig - **1 2 3 4 5 6** - Veldig enig

Q2: Jeg forsto raskt alle funksjonene i verktøyet

Sterkt uenig - **1 2 3 4 5 6** - Veldig enig

Q3: Knappene var der jeg forventet å finne dem

Sterkt uenig - **1 2 3 4 5 6** - Veldig enig

Q4: Mitt helhetsinntrykk av verktøyet er:

Veldig negativt - **1 2 3 4 5 6** - Veldig positivt

Q5: Kan du nevne to ting som du syntes er bra med verktøyet?

Q6: Kan du nevne to ting du ikke likte ved verktøyet?

Skriv gjerne ekstra kommentarer på baksiden av dette arket.
Takk for at din deltagelse i denne undersøkelsen.