

UI for Remote Operation of Drones

Designing a user interface for simultaneously operating multiple automated drones remotely in support of sea rescue missions

Master's thesis in Interaction Design and Technologies

Alicia Holmqvist
Stina Olsson

MASTER'S THESIS 2024

UI for Remote Operation of Drones

Designing a user interface for simultaneously
operating multiple automated drones remotely
in support of sea rescue missions

Alicia Holmqvist

Stina Olsson



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Division of Interaction Design
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2024

UI for Remote Operation of Drones
Designing a user interface for simultaneously
operating multiple automated drones remotely
in support of sea rescue missions
ALICIA HOLMQVIST
STINA OLSSON

© ALICIA HOLMQVIST, 2024

© STINA OLSSON, 2024

Supervisor: Mafalda Samuelsson Gamboa, Computer Science and Engineering

Advisor: Fredrik Falkman, Swedish Sea Rescue Society

Examiner: Staffan Björk, Computer Science and Engineering

Master's Thesis 2024

Department of Computer Science and Engineering

Division of Interaction Design

Chalmers University of Technology and University of Gothenburg

SE-412 96 Gothenburg

Telephone +46 31 772 1000

Typeset in L^AT_EX
Gothenburg, Sweden 2024

UI for Remote Operation of Drones

Designing a user interface for simultaneously
operating multiple automated drones remotely
in support of sea rescue missions

ALICIA HOLMQVIST

STINA OLSSON

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

Abstract

Drones are increasingly used as tools across various domains, and the Swedish Sea Rescue Society (SSRS) sees potential in utilizing them to aid sea rescue missions. Their Eyes-On-Scene project aims to streamline their rescue missions by deploying a drone immediately upon receiving an alarm, providing real-time footage of the site to the crew who can prepare according to the situation. In cases where multiple alarms are received simultaneously, it would be advantageous if a single remote operator could manage multiple drones – a possibility explored through this thesis. In response, the project followed an iterative design and evaluation process, incorporating interaction design methodology within phases of research, analysis, synthesis, and realisation. Findings from the process resulted in a set of nine key considerations, highlighting factors of user control, customisability and personalisation, cognitive load, situational awareness, and system transparency and feedback. The considerations are a product of the design process, which additionally produced a high-fidelity prototype of an interface adhering to these principles. The prototype is paired with annotations and is a direct deliverable to SSRS as well as a contribution to design practitioners. Additionally, this thesis provides valuable insights for researching or developing a multi-control drone UI in a sea rescue context. Potential future work is discussed in the report, highlighting the need to further evaluate the considerations.

Keywords: Interface Design, Drone, HDI, Interaction Design, Multi-control, GUI.

Acknowledgements

We want to thank Fredrik Falkman and the Swedish Sea Rescue Society for providing an interesting thesis project with the opportunity to contribute to a very important mission. A special thanks go to our supervisor, Mafalda Samuelsson-Gamboa, for the guidance and appreciated nitpicking throughout the process. We would also like to thank everyone who participated in the interviews, workshops and evaluations for their time and valuable feedback.

Alicia Holmqvist, Gothenburg, 2024-06-10
Stina Olsson, Gothenburg, 2024-06-10

List of Acronyms

BVLOS Beyond Visual Sine of Sight

DOS Drone Operation Software

EOS Eyes-On-Scene

GUI Graphical User Interface

HCI Human-Computer Interaction

HDI Human-Drone Interaction

HMI Human-Machine Interaction

HRI Human-Robot Interaction

HTA Hierarchical Task Analysis

JRCC Joint Rescue Co-ordination Centre

LOD Level Of Detail

NOTAM Notice to Airmen (aeronautical information)

PHEA Predictive human error analysis

POI Point Of Interest

RtD Research through design

SAR Search and rescue

SSRS Swedish Sea Rescue Society

UAV Unmanned Aerial Vehicle

UX User Experience

WP Waypoint



Contents

List of Figures	xv
List of Tables	xix
1 Introduction	1
1.1 Aim	1
1.2 Research Question	2
1.3 Stakeholders	2
1.3.1 Chalmers University of Technology	2
1.3.2 Swedish Sea Rescue Society	2
1.3.3 Practitioners in Interaction Design	3
1.4 Delimitations	3
2 Background	5
2.1 Drones	5
2.1.1 Current Swedish Regulations	6
2.2 Swedish Sea Rescue Society and The Drone Project	6
2.3 Related Work	8
2.4 Research Opportunity	9
3 Theory	11
3.1 Interaction Design	11
3.1.1 User Experience and Usability	11
3.2 Design Process	12
3.2.1 The Design Innovation Process	12
3.2.2 Research Through Design	14
3.3 Human-Drone Interaction	14
3.3.1 Drone User Interface	14
4 Methodology	17
4.1 Research	17
4.1.1 Literature review	17
4.1.2 Benchmarking	18
4.1.3 Interviews	18
4.1.4 Observation	19

4.2	Analysis	19
4.2.1	Affinity Diagram	19
4.2.2	Storyboarding	20
4.2.3	Scenarios	20
4.2.4	User Profiles	20
4.3	Synthesis	20
4.3.1	Brainstorming	21
4.3.2	Co-design Workshops	21
4.3.3	Morphological Matrix	21
4.3.4	User Flow Diagram	21
4.3.5	Wireframing and Wireflow	21
4.3.6	Prototyping	22
4.3.7	Heuristic Evaluation	22
4.3.8	HTA & PUEA	23
4.4	Realisation	23
4.4.1	Usability Testing	24
4.4.2	Semantic Differential Scale	24
4.4.3	Annotated Portfolios	25
5	Process	27
5.1	Literature review	27
5.2	Meetings with the client	27
5.3	Scenarios	30
5.4	Benchmarking	32
5.5	Semi-structured Interviews and Observations	33
5.5.1	Operational context	34
5.5.2	Regulations	35
5.5.3	System	36
5.5.4	Interface	36
5.5.5	Stressors and Human Factors	38
5.6	Table of Requirements	39
5.7	Wireframes & Wireflow	41
5.8	Workshop	44
5.9	Predictive Use Error Analysis (PUEA)	48
5.10	Prototyping	51
5.11	Heuristic Evaluation	53
5.12	Qualitative Usability Testing	57
5.12.1	Planning	58
5.12.2	Protocol	59
5.12.3	Results	60
6	Result	65
6.1	Considerations for the UI	65
6.2	The Prototype	68
6.2.1	Home screen	69
6.2.2	Planning Route	71
6.2.3	Operating One Drone	74

6.2.4	Monitoring Multiple Drones	77
7	Discussion	83
7.1	Execution and Process	83
7.2	Ethical Considerations	86
7.2.1	Informed Participant Consent	86
7.2.2	Accessibility	86
7.2.3	Research Contributions	87
7.3	Results	88
7.3.1	The Considerations and UX Factors	88
7.3.2	The Design	89
7.4	Future work	90
8	Conclusion	93
	Bibliography	95
A	Project Plan	I
B	Identified Design Patterns from Benchmarking	III
C	Interview Template	V
D	PUEA Table	IX
E	Consent Form	XI
F	Matrices of Results from PUEA	XV
G	User Evaluation Reference Guide	XIX
H	Answering Time – Rapid Questions in User Evaluation	XXI

List of Figures

2.1	The drone developed through the EOS project along with the developed launcher (catapult)	8
3.1	Illustration of the double diamond design process.	12
3.2	Visualisation of the design innovation process, with annotations of the seven modes of design.	13
3.3	Examples of principles to work by when utilising the design innovation process.	13
5.1	Scenarios covering seven different drone scenarios. Each scenario is captioned in bold text.	31
5.2	Photograph of the flight monitor used at Landvetter approach control	37
5.3	Perceived focus levels for monitoring flights in different ways.	38
5.4	Four frames of wireframe concepts 1a (top frames) and 1b (bottom frames), developed through brainwriting using Figma. The right frames illustrate each concept's layout for overviewing multiple drones. The left frames illustrate a full screen view of selecting a drone.	41
5.5	Two frames of the first wireframe concept that emerged by combining features from concept 1a and 1b (Figure 5.4). Numbers in the image mark particular elements and are referred to in the text.	42
5.6	First draft wireframes splitting the screen when monitoring a) two drones and b) four drones.	43
5.7	First draft wireframes of the overview layout when monitoring four drones. a) Two drone cards are collapsed. b) No drone cards are collapsed creating a scrollable frame.	43
5.8	Second draft of wireframes, with numbers referring to updated features described in the text.	44
5.9	Feature for determining the optimal time at scene in combination with time to the target. The border radius of the circle represents how far the drone can travel with the current configuration of ETA and ETS.	44
5.10	One example of the wireframes presented at the workshop, enhanced with highlighted areas, colours, and cursor.	46
5.11	An overview of the component library along with building blocks for the interface.	52

5.12	The card in this figure shows how the user can change the drone speed from the fastest to a custom speed with radio buttons, to enable greater endurance for the drone.	53
5.13	List of waypoints during a flight. The WP currently headed to is highlighted in white.	53
5.14	Tab bar with visuals and text showing the status of the flight	54
5.15	Navigation rails compared	55
5.16	Altitude schematics component indicating a risk of collision ahead by marking the route red.	55
5.17	A section of the overview. The left column shows how customisation facilities appear while hovering over it. Lines indicate that the size of each window can be adjusted and the plus-icon allows the user to add other modules.	56
5.18	The dialogue that appears when the connection to a drone is lost is to help recognise and recover from an error.	57
5.19	Before selecting a drone the user is presented with useful information about the drones in order for them to make an informed decision about which drone to select. The information is presented in the frame close to the drone's position.	59
5.20	In the overview, the user finds information about the drones and their routes. When asking rapid questions to the evaluation participant, all answers could be found in this row of cards.	60
5.21	A chart of one participant's mean answering time to the questions in relation to the number of drones they were operating. The first dot represents two drones simultaneously and the last dot represents seven drones simultaneously.	62
6.1	Home screen, the user is on its way to insert received coordinates to find the location on the map. Numbers in the image are referred to in the text.	69
6.2	Image of map settings and corresponding elements in the map.	70
6.3	The targeted coordinates are showing on the map as a red dot with an animated red ring, representing "sending out signals".	70
6.4	The inserted coordinates reveal a target on the map, showing as a red rot with an animated ring (as seen in Figure 6.3). The user has in this case two available drones to choose from to reach the target. They can select the most suitable drone for the mission.	70
6.5	The selected drone changes colour to orange.	71
6.6	Visualisation of how the system informs the user about inappropriate routes.	71
6.7	Module on the left side of the planning frame has been zoomed in to demonstrate how the user can adjust the speed in a comprehensible manner.	72
6.8	To create an area to do a search pattern, the user clicks the WP (a) and selects <i>Search area</i> (b).	73

6.9	Altitude schematics, alerting the user if there is a collision risk. The green line represents, in this figure, land and the red line is the section of the route which is in collision course.	73
6.10	Altitude schematics visualising a clear course.	73
6.11	The launch is activated on a long-press on the launch button. To provide the user with feedback that the button needs to be pressed and held, an orange bar fills the button from left to right.	74
6.12	The frame during an active flight. The user sees the drone's position in the map and its altitude in the altitude schematics. The camera on the drone's image is visible in the bottom right corner. Numbers in the image are referred to in the text.	75
6.13	When the drone needs to make way for other vehicles, one option is to immediately start loitering. The image shows how the interface informs the user that the drone is loitering.	75
6.14	When a boat or other vehicle appears in the camera's view, AI can identify the vehicle and inform the user that the air space above it should be avoided. It is visualised through a yellow cylinder in the frame.	76
6.15	The symbols on the map change their appearance based on the obstacles' height or altitude. The more transparent red aircraft transponder indicates that the vehicle is at a high altitude. The user can therefore devote less attention to it. By hovering on the symbol, the user is provided with detailed information.	77
6.16	The overview when the user has two active flights. The information about each drone is collected in columns with corresponding tabs, which the user can click to view a larger map (Figure 6.12) and more detail about the drone and the environment.	78
6.17	Each flight's information is collected in columns, and each type of data is presented in rows. The user can choose to focus on one flight, or one type of data (flight data, map, or camera) by checking columns or rows.	78
6.18	A tab leads the user to a more detailed view of a specific flight. The tab itself also contains information in a low level of detail (name of the drone, and estimated time to next waypoint).	78
6.19	The tabs are collected in a tab bar. In the figure, the left tab is highlighted in red, to indicate that the flight needs the user's attention. By highlighting a tab, the notice can be seen from anywhere in the interface.	79
6.20	When connection is lost, the drone initiates a fail-safe mode, which the user will be informed about. The system suggests actions the drone can take when finding connection and the user can consent to one of them.	79
6.21	If the user wants to add a module, a text box is one of the modules that can be added to the overview.	79
6.22	The overview when flying seven drones, the maximum amount of flights to monitor in a column view.	80

6.23	When flying eight or more drones, the overview changes structure from columns to a grid with two rows. The maximum amount of flights is ten.	80
7.1	Illustration of our path in the Design Innovation Process.	85
A.1	Gantt schedule over thesis project plan	II

List of Tables

5.1	Table of requirements divided into six categories. Each requirement is ranked in priority from could, should and must have.	40
5.2	Workshop participants	45

1

Introduction

The following chapter presents the aim and purpose of the thesis, as well as the research question, stakeholders and delimitations.

1.1 Aim

In 2023, 1017 calls for rescue at sea was received by the Swedish Maritime Administration, out of which 17 % were emergency calls [1]. A way to streamline, and make rescue missions more cost- and environmentally effective is to deploy camera-equipped drones as the “first responders” – an ongoing project within the Swedish Sea Rescue Society (SSRS), called the Eyes-On-Scene (EOS) project. The images captured by the drone can assist the maritime rescuers in determining the appropriate type of boat as well as the number of rescuers needed for the mission.

The aim of this thesis is to design a prototype of a web based interface for operating such drones, setting the baseline for further development of the interface that will be implemented in Swedish sea rescue missions. Particular focus will be on enabling one drone pilot to manage multiple drones in different geographical locations and that they are operated efficiently and safely to support sea rescue missions. To achieve efficiency and safety, it should be easy to use and the pilot should feel in control of all drones they are flying, in order to assist the sea rescue crew proficiently.

Furthermore, the project aims to contribute to the research within human-drone interaction (HDI), more specifically interaction with interfaces for remote- and multi-controlling. While this project is focusing on drones in a sea rescuing context, the findings could be applicable in related areas as well.

1.2 Research Question

Given the above, the thesis aims to answer the following questions:

RQ: What should be considered when designing an interface for simultaneously operating multiple drones efficiently and safely in a sea rescue context?

Supported by the following sub-questions:

SQ1: What user experience factors are particularly important in the simultaneous operation of multiple drones in a sea rescue context?

SQ2: What solutions should be implemented in an interface to address the framework of design considerations?

1.3 Stakeholders

This section will follow by identifying the major stakeholders of the project.

1.3.1 Chalmers University of Technology

Chalmers University of Technology is the university where the master's thesis will be written thus provides the requirements and guidelines which the thesis will have to fulfil in order to be approved. Chalmers provides an examiner, Staffan Björk, approving and grading the thesis, complemented by an academic supervisor, Mafalda Samuelsson-Gamboa, responsible for overseeing the academic aspects and learning outcomes. The master's programme *Interaction Design and Technologies (MPIDE)* teaches a multitude of skills to bring an idea from needs and requirements through research and ideation into interactive prototypes ready for evaluation. The courses provide different approaches towards interaction design. Skills gained from the courses *Graphical Interfaces (DAT595)*, *Emerging trends (DAT540)*, and *Information visualisation (CIU187)* will be utilised to meet the requirement of involving significant specialisation within the thesis work in relation to MPIDE.

1.3.2 Swedish Sea Rescue Society

As stated in section 1.1, SSRS is currently researching the use of drones in rescue missions. A project called Eyes-On-Scene (EOS) The emergence of this thesis brief was a result of the EOS-project, making SSRS the client of this thesis.

The main user group for this project are the drone pilots at SSRS, which will operate remotely. Currently, no such position exists within SSRS but for the sake of the project, a potential future user group will be recruited within their member squad. After discussions with SSRS regarding the responsibilities and the types of tasks that the pilot is most likely to perform, the people envisioned as drone pilots are

those who have some previous experience with drone or flight operations, such as private, or commercial pilots as well as air traffic controllers. This group already has knowledge of airspace regulations. In addition, air traffic controllers are used to multitasking in their profession. Going forward, it is anticipated that some form of training will be required for future drone pilots. Others affected by the quality of the interface are the software developers, who need to be able to implement the final design in a reasonable way, and the rescue crew, who depend on receiving valuable footage in time.

1.3.3 Practitioners in Interaction Design

This thesis is expected to result in an interactive prototype of a drone user interface. The goal of the interface is to maintain a high level of cognitive efficiency, with sub-targets of high usability, situational awareness, and system transparency. The project outcomes, including process and research, are hoped to be useful for practitioners within interaction design, drone development, or search and rescue (SAR) projects, especially those interested in multi-control factors.

1.4 Delimitations

The design will not be developed into a fully functional prototype during this project, which should be taken into account when analysing the evaluation results. Furthermore, the interactive prototype will not be evaluated in an actual drone flight environment, meaning that potential errors that could occur during a rescue mission may be overlooked.

In the course of the project, a user group with a flight-related background was recruited from SSRS volunteers. However, SSRS has not yet established specific criteria for selecting future drone pilots, including aspects such as employment status, background, or number of individuals involved. Therefore, the user group identified for the project may not definitively represent the ultimate end-user, but it is our current best estimate.

Although drones are becoming more familiar to society, current regulations do not adequately support the use of drones. Within the EOS-project, a lot of the work done until now has revolved around adapting to regulations and filing for certificates in order to even get to perform test flights, see Section 2.1.1. For the scope of this thesis, it is assumed that necessary regulations are in place, allowing SSRS's drone project to be realised. Only regulations that directly affect the interface design will be considered.

2

Background

This chapter will contextualise the work by further describing the field of drones, and the Swedish Sea Rescue Society, followed by presenting related work as well as a description of the research opportunity for this project.

2.1 Drones

According to Miron, Whetham, Auzanneau *et al.* [2] drones can be described as “remotely piloted aircraft of varying sizes”. A drone is always unmanned and the majority are equipped with cameras. There are many different types of drones and they can be flown with autopilot or manually, they are however always dependent on a drone pilot who has the ultimate responsibility of the vehicle. The most common type is multi-rotor drones which have several horizontal propellers that enable the drone to hover in a static position, take off without external equipment, and land serenely. Drones can also have fixed wings, referred to as fixed-winged drones. These use vertical propellers instead, which, when not needed, can be deactivated to let the drone glide using the wings. Unlike the multi-rotor drone, fixed-winged drones can not hover, but to imitate a similar action they circulate, or loiter, around the targeted coordinates. Variants and combinations of the two also exist.

There are many fields in which drones are used, including the agriculture industry, photography and filming, public services such as police and fire brigades, and transportation of goods. Drones have also shown to be cost efficient and a helpful tool for SAR missions worldwide [3]–[7]. The Norwegian Sea Rescue Society pioneered in 2014 when they received their first rescue drone. Norway has over 25 000 km of coastline, including islands, and fjords – a demanding environment at sea with challenging weather conditions. The drones have proven a useful tool to make rescue missions more efficient by for example locating the exact position to the crew, sending footage to the crew to determine the best route and equipment, and an infrared camera for foggy weather conditions [8]. Another benefit of drones was highlighted in Nafiz, Khan and Neustaedter’s study [9] where drones were being explored to assist firefighters. It was clear that the sight of a rescuing drone can be a stress reliever for the distressed person, as a reassurance that help is on the way.

2.1.1 Current Swedish Regulations

As mentioned in Section 1.4, Swedish regulations are currently not supporting the use of drones for the purpose of emergency situations. To fly a commercial drone, it must be within the user's line of sight. Current Swedish regulations for flying drones beyond visual line of sight (BVLOS) however, requires specific permissions. Firstly, a permanent licence is needed. Secondly, a permission to fly in someone else's airspace needs to be requested and reported weeks ahead. After that follows a long list of additional tasks to complete before being able to fly [10], while hoping that the weather forecast keeps clear and calm for the planned day of flying. For SSRS's drone project, it is necessary to be able to fly BVLOS at unannounced times.

Regulations that are important for the interface to assist the operator with are areas where the drone is not allowed to fly. Examples of this are in restricted areas, such as airports' air space, over inhabited land, and temporary military airspace. Further, the drone operator should in some way separate from humans, manned aircraft, and ships. Additionally, SSRS's drones are permitted to fly at a maximum altitude of 45 metres above sea level.

Another regulation to take into account during this project is that a drone operator holds the ultimate responsibility, and therefore, even if the drone flies with an auto-pilot, it is important that the operator also makes the ultimate decisions about the flight.

2.2 Swedish Sea Rescue Society and The Drone Project

This thesis project was conducted together with SSRS. They are a non-profit, non-governmental SAR organisation that has provided sea rescue at Swedish coasts and lakes since 1907. SSRS is solely funded by membership fees and private donations, and relies on over 2400 volunteers, out of which three hundred are on call at any given time to carry out their mission: *No one should die at sea.*

Through an agreement with the Swedish Maritime Administration, SSRS is part of the nationally organised sea rescue and is involved in the majority of all sea-related emergency calls that reach the Joint Rescue Co-ordination Centre (JRCC) [11]. They have 74 stations located across Swedish coasts and major lakes, along with 260 modern rescue vessels, making it possible to reach their goal of departing within 15 minutes or less from the time an alarm is received [12]. The purpose of SSRS, under agreement with the Swedish Maritime Administration, is to:

1. Increase interest in maritime search and rescue.
2. Suggest measures to improve maritime search and rescue.
3. Perform search and rescue operations in Swedish waters.

The second purpose includes investment in research and development projects, one of which is the Eyes on Scene (EOS) project (the project from which the scope of this thesis was derived), led by Fredrik Falkman, head of innovation at SSRS.

Currently, when an emergency call is received, on-call members first travel to the station, which takes 8 minutes on average, gather mission-appropriate equipment and generally leave within 12 minutes of the call. The calls that reach SSRS have been categorised into *Life threatening*, *Can become life threatening*, *Rapid medical care*, and *Oil pollution*, all of which require different routines for the crew [12]. However, the calls that reach SSRS often lack contextual information, making it difficult to determine the level of emergency. This uncertainty means that the crew prepares for the worst and might choose the gas-powered boat instead of the short-range electric boat, or spends time gathering excessive supplies for the mission. To address this issue, the EOS project started. Fixed-winged drones have been developed along with a launcher (a catapult) that sends the drone for take off (Figure 2.1). The idea is that the drone will rapidly head to the location of the emergency. Upon reaching the location, the drone collects footage of the scene, which is then transmitted to the crew at the rescue station, allowing them to view the situation and prepare accordingly. In some cases, the drone may be needed to assist in search missions. When the mission is complete, the plan is to land the drone in the water near a rescue boat on site for retrieval.

Currently, SSRS is working on an early prototype of the drone's operating system. This system will implement a one-to-one control relationship between a pilot and a drone. However, there will be times when the number of emergency calls exceeds the number of available drone pilots. During summer in Sweden, especially in July, SSRS reaches its peak number of emergency calls due to the many recreational boats in use [11], [13]. This puts pressure on the number of volunteers available to operate one drone for each alarm, which is why SSRS is asking for a system where one pilot can operate several drones.

The task for this thesis is to design an interface for the drone operating system. While a design for an interface with all aspects considered is asked for, a particular focus will be on exploring the possibilities for one pilot to operate multiple drones at different locations at the same time. This would mean that one pilot could operate drones in the north of Sweden while also operating drones in the south. Hopefully, this will minimise the human resources needed in times where drones are called out simultaneously, without compromising safety.

To summarise, the drones will provide the rescue team with an overview of the situation, allowing them to prepare more quickly and accurately, reducing the departure time without compromising safety. It is also hoped that the ability to determine the nature of the mission will enable more economical and sustainable decisions within the SSRS organisation.



Figure 2.1: The drone developed through the EOS project along with the developed launcher (catapult)

2.3 Related Work

Langerwisch, Wittmann, Thamke *et al.* [14] conducted a field study on implementing a graphical user interface (GUI) for multi-controlling drones and unmanned ground vehicles for surveillance purposes. Two GUIs were tested, one on a tablet and the other on a PC. Both displayed an aerial map of the surveillance area, including the locations of all vehicles. Two commands were tested in the study: Move to a defined destination and Observe a Point of Interest. All four ground vehicles were able to move autonomously to a desired destination, and all drones were able to loiter autonomously around a point of interest and position themselves in its proximity.

Building on the topic of multi-robot control, Niroui, Liu, Bichay *et al.* [15] investigated the key considerations for designing a graphical user interface for multi-robot control in urban SAR environments. The GUI enables the operator to pre-configure rescue robots into search teams and modify the teams during the mission if necessary. Additionally, the operator can alternate between monitoring the entire team and controlling each robot individually during a search mission. The findings highlight the significance of considering situational awareness and reducing cognitive load for operators when designing interfaces for multi-robot control.

Moreover, Ruff, Narayanan and Draper [16] investigated the impact of different autonomy levels on users operating multiple drones. Their evaluations involved up to four drones remotely operated by a single user, each exhibiting different autonomy levels adjusted through interfaces featuring manual control, management-by-consent, and management-by-exception. Variables measured were mission efficiency, percentage correct detection of incorrect decision aids, workload, situation awareness ratings, and trust. The results indicated that an automation level integrating management-by-consent is to be preferred.

In addition to the above, Wang [17] emphasises the importance of establishing trust between users and autonomous technology. The key factor in managing this trust, according to the authors, lies in the level of detail (LOD) of the user interface. In their study, interfaces with alternating LOD's were evaluated on a drone system for SAR missions. A higher LOD exposes less information, facilitating an easier overview of key details, but potentially at the expense of concealing vital information. Ultimately, the level of detail should be considered in interface design as it can increase transparency and confidence in the system, avoiding unintended consequences on comprehension, task performance and workload, and thereby increase the overall effectiveness and user experience.

2.4 Research Opportunity

The field of drone research is rapidly advancing. Research relevant to this project that were found are studies on drones used in SAR missions, drone swarms, sweeping, and other closely related areas. However, no published material specifically on considerations when designing interfaces for monitoring multiple SAR drones situated at different locations could be found. The project was requested by SSRS after identifying a need of such an interface.

As SSRS's drones are planned to be used for emergency situations, their flights, routes, and additional permissions can not be planned ahead, which is now needed according to current regulations (see Section 2.1.1). The route needs to be created on-the-fly when emergency calls are received. In this specific area of research, no publication on this has been found.

3

Theory

This chapter will cover theories related to the design research and specific research connected to this thesis project.

3.1 Interaction Design

Interaction design can be described as the design of interactive products and services in which the designer’s focus goes beyond the item being developed to include the way users will interact with it. There is a strong emphasis on uncovering user needs, limitations and contexts, enabling designers to tailor output to meet requirements [18]. Interaction design incorporates elements from various disciplines, such as cognitive science and information architecture. Although not interchangeable, the fields of Human-Computer Interaction (HCI) and User Experience (UX) design contribute significantly as closely related domains to Interaction Design.

3.1.1 User Experience and Usability

As Interaction Design is very much about designing for the user, aspects of UX and usability are central to the design of an interactive user interface. While the terms UX and usability often overlap and are sometimes used interchangeably, they serve different purposes within Interaction Design.

User experience refers to how a user experiences a product or service. It raises questions such as what emotions arise during use and why. As stated by Sharp, Preece and Rogers [19], user experience can not be designed, but it can be designed for. Users will experience situations differently, but certain UX goals can be articulated and optimised in line with the designer’s intentions. By considering factors such as cognitive load, the designer is better able to achieve a satisfactory UX. In general, desirable goals include satisfaction, helpfulness, engagement etc. Conversely, feelings of frustration, boredom and discomfort are examples of undesirable UX [19]. Overall, these emotional aspects of a product are referred to as hedonic aspects, which aim to please the user [20]. Usability aspects, on the other hand, are according to Hassenzahl [20] referred to as pragmatic attributes: “a product’s perceived ability to support the effective and efficient achievement of tasks”. Due to their subjective

nature, UX metrics are more difficult to evaluate compared to the objective measures of usability, such as Nielsen’s usability heuristics (Section 4.3.7). Nevertheless, there is a strong correlation between the two, as a product with high usability tends to provide users with a positive user experience.

3.2 Design Process

Following a design process is valuable for structuring the design work and there are various models aiming to do so. The double diamond (Figure 3.1) is a popular approach to the interaction design process and we believe that it is a good way of learning design thinking. However, the design innovation process, described in the following section, opens up more possibilities for agile design work. Since a design process takes shape after hand depending on the results from each step, the design innovation process is more in line with how we expect this thesis to unfold.

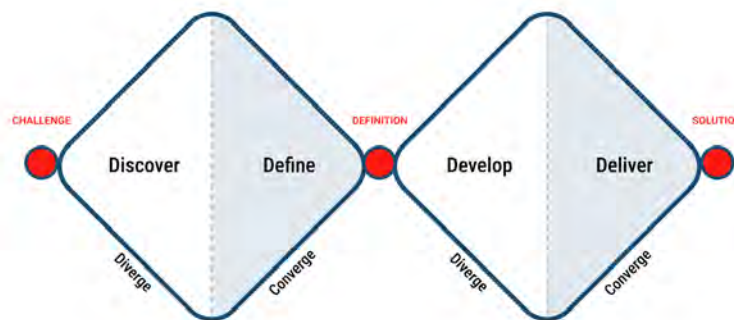


Figure 3.1: Illustration of the double diamond design process.

3.2.1 The Design Innovation Process

The design innovation process is made to promote innovation within an organisation and consists of seven modes of design (*Sense intent, Know context, Know people, Frame insights, Explore concepts, Frame solutions, and Realise offerings*) divided into four different phases: *Research, Analysis, Synthesis, and Realisation* (Figure 3.2). Kumar [21] also states that the design innovation process is “oscillating between poles of *Real* versus *Abstract* and *Understanding* versus *Making*” which has also been added into the gaps of the 2 by 2 grid [21]. The process encourages the designer to iterate (Figure 3.3a), and follow a non-linear design process (Figure 3.3b) as the seven modes are not phases but rather different approaches to what focus the designer has at the moment. The process also allows the designer to revisit the first mode during the process or as a last step to connect the result to the purpose.

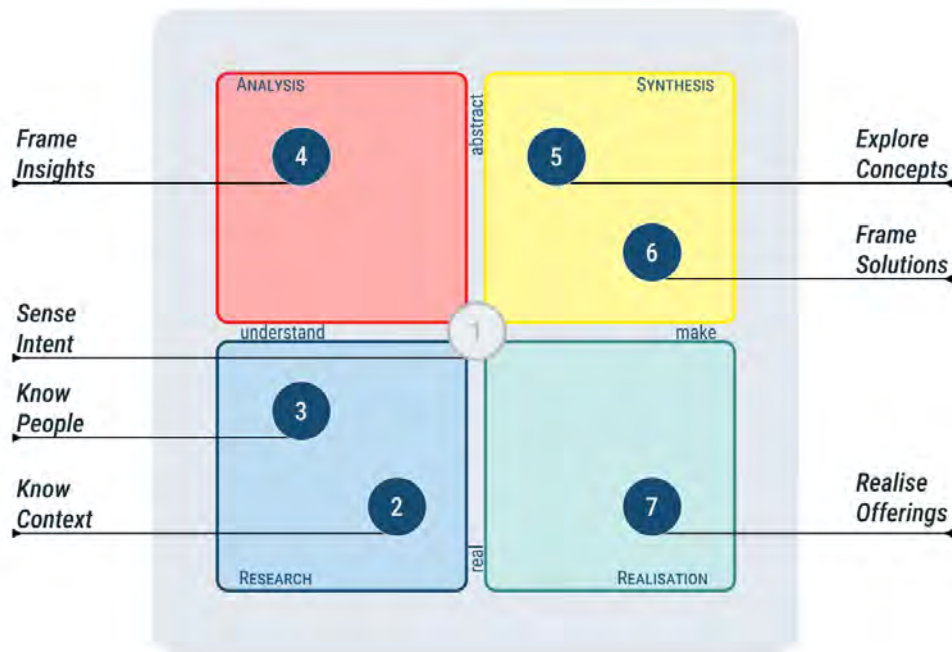


Figure 3.2: Visualisation of the design innovation process, with annotations of the seven modes of design.



(a) The design innovation process with an iterative path.

(b) The design innovation process with a non-linear path.

Figure 3.3: Examples of principles to work by when utilising the design innovation process.

3.2.2 Research Through Design

Research through Design (RtD) is an approach, and term first coined by Frayling [22] as a way of bridging the gap between research and design practice, meaning that research used to be a term to describe reviews of old papers rather than imposing that a piece of art or design likewise can produce scientific contributions. As of now, RtD is a recognised approach within design and HCI practices to conduct scholarly research that employs design methods and processes in a reflective way and with the intention of generating new knowledge [23]. Designers use their work, including sketches, prototypes and models, to interrogate ideas, test hypotheses and raise new questions for future work. It is constituted as such that recordings of the process, documentation and critical thinking should promote the design process to iterations [24]. Products of the RtD approach may be contributions in the form of design guidelines, future work proposals or design concepts. In addition, Gaver [25] argue that one of the valuable roles of design theory is to make the design space of artefacts accessible to the design field, and propose portfolios as a way of bridging this research gap. In particular, Gaver [25] recognises annotated portfolios as a valuable contribution because “if artefacts embody theory, they do not encode it”, emphasising the importance of documenting design as research.

3.3 Human-Drone Interaction

Human-Drone Interaction (HDI) has become a field within Human-Robot Interaction (HRI) which in turn has its roots in the well-established Human-Machine Interaction (HMI). As technology evolves, research follows along, leading to more specified fields within interaction design. No longer is the term HMI enough to cover the network of interactive technologies people encounter in their everyday lives. This thesis will focus on HDI but the connections to parallel fields will be a source of inspiration.

HDI can be defined as the study of understanding, designing, and evaluating drone systems for use by or with human users. When summarising current research, HDI can be categorised into four major fields; creating new control modalities, enhancing human-drone communication, evaluating interaction distance, and developing new use cases [26]. The second and the third category mentioned, circle around the interaction between the drone and the humans who encounter it, not operating it. Essentially, these aspects come down to the societal acceptance of drones in different scenarios. Nonetheless, this thesis will focus on the first mentioned category which includes research on how to develop a user interface for operating drones.

3.3.1 Drone User Interface

Drones can be steered using external or internal piloting. External piloting refers to the conventional hand controller used in many hobby drones that operate within line of sight, while internal piloting refers to a first-person view of the pilot simulating the feel of sitting in an onboard cockpit [27]. Most commonly, the internal

piloting approach is using some type of Graphical User Interface (GUI), whether it be on a desktop environment, a mobile or a specialised simulator setup. Controlling drones via a GUI offers notable advantages in practical aspects such as vehicle setup, mission monitoring, manual handling with straightforward movements, and data collection. These functionalities are facilitated by standard components in the interface, including a control panel, parametric view, and vehicle camera view, which in the domain of UI design are referred to as design patterns. These patterns act as design templates, enabling designers to select the most effective of conventional solutions to target a specific interface problem and tailor it to the context. Design patterns can be components, such as a navigation bar, or strategies employed in the interface, such as highlighting all primary buttons [28].

4

Methodology

The following chapter displays a collection of relevant methodology to be considered within the design process of this thesis. To make use of methods in a fruitful way, methods should be carefully considered for what purpose and when in the design process they should be used. Because of a design process' non-linear nature, the Design Innovation Process [21], presented in section 3.2.1, was chosen to follow. Methods have been categorised within each of the four quadrants *Research* (4.1), *Analysis* (4.2), *Synthesis* (4.3), and *Realisation* (4.4). A plan was made outlining when the methods would be used. The project plan can be found in Appendix A.

4.1 Research

In the bottom left quadrant of the design innovation process lies the two *Research* modes; *know context*, and *know people*. It is framed with the words *real*, and *understand* because the goal is to get a real and tangible understanding of the user and the context by for example interviews or observations.

4.1.1 Literature review

Literature reviews are made to research and receive information about fields related to the project. By converging information from different sources, the literature review should provide the current project with relevant conclusions within the field. To find relevant literature, it is helpful to pinpoint topics and keywords to search for. Literature includes, but is not limited to, books, journals, conference proceedings, theses, documented design projects, and reports. For the review it is not only important to select relevant literature, but also that the sources are credible. Literature also includes blog posts and websites. The latter types of sources of information should be used with extra caution, as they have not been peer-reviewed for credibility. The review should be concluded with clear suggestions of how it has informed the current project and how the information can be used in the progression of the project [24].

4.1.2 Benchmarking

UX benchmarking is a quantitative method for comparison between a design with an earlier version, competitors, or to track the overall experience and progress of a product. For an end-to-end benchmarking analysis, it is conducted in seven steps:

1. Choose what to measure
2. Choose how to measure
3. Collect first measurements
4. *Redesign the product*
5. *Collect more measurements*
6. *Interpret findings*
7. *Calculate return on investment*

The first step is to define what type of product is going to be measured (website, application etc.), who the user is, and what task(s) to focus on. To be able to conduct the benchmarking, it is of high importance that the key metrics are measurable and that they reflect the values that are interesting for the project. Secondly, how to measure the metrics should be decided. The most common way to measure is through analytics but usability testing, and surveys are some other options. The methods used should however be in line with what to measure and what other information is already accessible [29]. The first time measurements are collected, a baseline is set. For a competitor benchmarking, an aspect of the product can be compared with a function of a competitor's product with the same or similar outcome [29]. When comparing graphical user interfaces, it is valuable to map out what design patterns (Section 3.3) exist, what functionalities they cover and in which context. For the research phase, this is how far the benchmarking goes. Redesign and collection of more measurements would be used for further iterations.

4.1.3 Interviews

Interviews belong to the category of survey research and is a way of collecting first-hand information from people in any part of the design process. The participants can vary from being novice users, field experts, or other stakeholders, where the goal is to get insights on their personal experience, opinions, attitudes, and perceptions of something. Interviews often vary between being of a structured, unstructured or semi-structured type. Structured interviews contain a pre-written script with questions, while unstructured interviews are more flexible and of a conversational nature. Unstructured interviews depend on the interviewers ability to steer the interview questions according to the relevance of the project. According to Martin and Hanington [24] this can be more comfortable for the participant but also more

sensitive to the interviewer’s bias, compromising the consistency and credibility among several interviews. While structured interviews are perceived as more formal they have the advantage of being easier to control, more time efficient and easier to analyse, especially in a greater quantity of interviews.

Laddering is a technique used in unstructured or semi-structured interviews to uncover the underlying reason for a particular statement by asking “why?” According to [30] it is easier for people to express their like or dislike of something than to identify the reasons why. Ultimately, laddering reveals how attributes affect each other and how they relate back to the overall perception of something.

4.1.4 Observation

Observations are a way of gaining insight into how people act in different situations and what aspects might influence their decisions.

Contextual observations can be conducted to collect information on how people act in different context since products and services are rarely used in isolated scenarios. By observing how people act in different contexts, valuable information can be gathered. Contextual observations focus on users’ expressions, feelings, and experience [30] and therefore rely on the designer’s interpretation of physical expressions.

4.2 Analysis

The upper left quadrant is for *Analysis* and consists of one mode: *frame insights*. This is aimed at defining guidelines and principles to work with for the upcoming modes, which is done with methods for sorting, screening, and prioritising data through the methods presented in this section.

4.2.1 Affinity Diagram

Creation of affinity diagram, also known as the KJ Method [31], is a method to sort, define, and prioritise data, and keywords gathered from observations, interviews, and user tests. These information gathering techniques generally generate qualitative data and affinity diagrams are used to translate the data to more measurable user needs. To analyse the data it is necessary to extract the useful insights from every method.

Affinity diagrams are traditionally made with sticky notes, where keywords, or other valuable takeaways from the session is written down. Nowadays, there are several digital tools to conduct the method such as Miro, FigJam (Figma), and Mural. Some digital tools also have AI-powered plug-ins for creating the diagram.

In a nutshell, the different sticky notes which relate with each other are grouped together to give a visual overview of what respondents have the most opinions about. The sticky notes can also be grouped within groups to create subcategories [24].

4.2.2 Storyboarding

Storyboards can be used to visualise an existing or envisioned situation. The existing situation should be described with real data, such as the results from observations. For an envisioned situation, storyboards are good for describing early ideas to potential users and communicating concepts to others [30].

A storyboard is a combination between text and images to make a timeline of happenings, and possible actions in a situation. It can be done with pen and paper or digitally [32].

The purpose of storyboarding is to present something to someone who has not been involved in developing the concept. When involving more people in the development phase, such as prospective users, in a workshop session or interviews, it is necessary that they understand what the aimed result, and the context, is to have an efficient session with valuable outcomes.

4.2.3 Scenarios

Scenarios serve as a method for designers to frame envisioned situations and contextualise ideas. By documenting these scenarios in a structured format, a timeline of events can be established, facilitating a comprehensive understanding of the progression of events and requirements. Furthermore, scenarios can be used to communicate various potential outcomes to stakeholders without the need for physical artefacts.

Often, scenarios are written from the perspective of a user. This approach connects design ideas to the people who will ultimately use the product, thereby focusing on the relationship between the user and the product [30].

4.2.4 User Profiles

User profiles is a technique used to understand and map variations within the target group. Key attributes of users, such as their occupation, income, skills, and attitudes, are identified based on the specific project. These attributes are then placed in a table to display the spectrum of user categories. This provides a comprehensive overview of potential users and their differences, giving an initial insight into their needs and requirements [30].

4.3 Synthesis

In *Synthesis*, in the upper right quadrant, are the two modes *explore concepts*, and *frame solutions*. This is where ideas are developed, compared, and sorted by internal assessment methods presented in this section.

4.3.1 Brainstorming

Brainstorming comes in many different sizes and shapes. Two examples of brainstorming methods are bodystorming, and brainwriting where bodystorming is a physical exercise and brainwriting is about writing and drawing ideas in silence and building on other participants' ideas. The outcome from this method should be a range of ideas generated by exercises to broaden the mind.

4.3.2 Co-design Workshops

Co-design workshops bring together users, stakeholders, and designers to rapidly and iteratively critique and build on design concepts. The aim is to ensure that the needs of the people being designed for remain at the centre of the design process and to inform the future direction of the design. Users and other stakeholders play an active role in the design process, rather than passively responding to design decisions. Ultimately, co-design promotes the idea of designing with the user rather than for them [30].

4.3.3 Morphological Matrix

The Morphological Matrix aims to divide a problem into smaller, more manageable pieces and find different solutions for each sub-element of a product or system. It consists of a table or grid with different categories, each representing a component of the problem. By systematically combining different elements from each category, new ideas can be generated. This method can be valuable in stimulating the creativity of the team and bringing out new perspectives on a possible solution. The solution space generated is often large and needs to be narrowed down to eliminate solutions that do not meet the product requirements [30].

4.3.4 User Flow Diagram

User flow diagrams are a tool for visualising the steps a user can take within an interface. They are valuable assets in UX design for determining the optimal ways of interacting with an interface that meets specific user needs. The flow chart can visually describe the interface structure and communicate design intentions to stakeholders and the design team using common geometric symbols. User flow charts can provide a solid foundation for developing wireflows and wireframes, as well as a guide for building user scenarios in user testing [18].

4.3.5 Wireframing and Wireflow

Wireframes are simple drawings to structure an interface. It is efficient because the designer is not obligated to nor should create high fidelity images. Due to the lack of imagery, font, and colours, the focus is directed toward functionality rather than aesthetics [30]. The wireframes represent each frame necessary to build the product. From wireframes, a wireflow can be created. It is a fast method and good for to visualise early prototypes of the use flow between each frame [30].

4.3.6 Prototyping

Prototypes within interaction design is a way of showing a concept and its interactivity. Unlike mock-ups and wireframes who are considered static, prototypes should act as an interactive representation [33]. An interactive prototype is a tool for design teams to make sure the design works as intended, to showcase the design to developers and other stakeholders, and to use as a template for user testing. Iterating based on the feedback received from an evaluation of an early prototype is an efficient way to address issues early in the process [34].

Prototypes can be categorised into different levels of fidelity, which refers to how closely the prototype reflects the final product. The fidelity level of an arbitrary prototype may vary in terms of its functionality, visual design, or content. An example of a low fidelity prototype in interaction design is a paper prototype, while a Figma prototype with colours, font selections, and interactivity often is referred to as high fidelity.

4.3.7 Heuristic Evaluation

Heuristic evaluation is a method used to detect usability issues with a user interface before introducing users into the evaluation process, pioneered by Nielsen and Molish [35]. The heuristics are a set of rules that can be seen as the baseline of usability within UI and is evaluated by experts or novice users. Nielsen defined a set of 10 usability heuristics that can be universally applied to gauge usability issues. While these work as a good starting point, heuristics should be carefully considered to reflect individual products and their context. Rather than making design decisions based on intuition, the defined heuristics should help point the design team to a common view on what need to be fixed [24].

1. **Visibility of System Status** is about transparency and providing user feedback.
2. **Match Between the System and Real World** is about keeping to conventions and real-world logic.
3. **User Control and Freedom** is about enabling undo, redo and exit actions.
4. **Consistency and Standards** is about keeping design elements to both internal and external conventions. Internal conventions refer to being consistent within the product.
5. **Error Prevention** could for instance mean making it difficult to perform a specific task or to provide strong alternatives.
6. **Recognition Rather than Recall** is about minimising the user's memory load. The user should not have to remember information from one part of the interface to another.

7. **Flexibility and Efficiency of Use** is directed to expert users and is meant to streamline the workflow by customizing their experience.
8. **Aesthetic and Minimalist Design** is about removing irrelevant aspects of features and components in the interface to minimise distraction and ensure focus on the primary goal.
9. **Help Users Recognize, Diagnose, and Recover from Errors** stresses the importance of presenting error messages in a way that will help users notice and recognize them.
10. **Help and Documentation** refers to when the user needs additional help in order to complete a task.

4.3.8 HTA & PUEA

Predictive Human Error Analysis (PHEA) is a method within HCI for identifying potential use errors and assessing their consequences in order to eliminate use errors prior to end-user evaluations [37]. PHEA involves systematically questioning, “What actions might users perform incorrectly?” and “What are the repercussions if such errors occur?” for each step within the interface. To carry out the PHEA, it is necessary to outline the tasks to be evaluated, for which a HTA is a useful method.

Hierarchical Task Analysis (HTA) is a method to dissect and comprehend the hierarchical organisation of tasks within a system. The analysis continues by breaking down the subordinates into further sub-steps until a sufficient level of detail is obtained. The primary goal of HTA is to create a detailed representation of tasks and their relationships, unfolding the step-by-step processes users undertake to achieve specific objectives. The HTA offers structure and context, facilitating further analysis with the PHEA. This, in turn, establishes a groundwork for redesign based on identified issues within the interface.

An elaborated version of PHEA is Predictive Use Error Analysis (PUEA), which Bligård and Osvalder describes as “Development of [Action Error Analysis], [Systematic Human Error Reduction and Prediction Approach] and PHEA to better predict, identify and present use errors” [38]. Findings result in matrices to judge errors, their probability and severeness.

4.4 Realisation

Realisation is where the seventh mode (*realise offerings*) is placed. This is when the ideas generated from the previous modes are evaluated at a detailed level. It is tested by the prospective end user to document the functionality, usability and the individual’s overall experience. Results of this phase may reveal design flaws or insights, leading to iteration of earlier phases until the finale of this phase; project solution.

4.4.1 Usability Testing

Usability testing is an evaluative method that involves different users to target usability matters in a system. The evaluation allows the design team to observe the systematic walk-through that users take to complete a set of specific tasks. The tests are designed around scenarios that contextualize the tasks, specifically designed to represent a desired end-user goal. According to Martin and Hanington, observers should aim to detect aspects of a specific task, goal, or feeling based on participant behaviour, such as:

- understands the task but takes too long to complete
- understands the goal but completes it with an alternative approach
- resigns from the task
- completes the wrong task (poorly defined goal)
- express extreme feeling (positive or negative)
- makes a suggestion for improvement.

In order for evaluators to interpret the participants' experience, usability tests often follow the format of Think-Aloud Protocol technique. The most common approach to think-aloud is where the participant works through the usability test while articulating their actions, thoughts and feelings through each step [24].

Before an evaluation, it is beneficial to conduct a pilot test to ensure that the prototype, interview questions and structure are well prepared for the evaluation.

4.4.2 Semantic Differential Scale

Semantic differentials are a set of contrasting words used to assess the expressive properties of a product or used as a discussion prompt for qualitative data gathering. The semantic differential scale exposes the outer limits of a product's semantic space and aims to assess participants' attitudes towards an object. Antonyms are usually chosen as the polar ends of the scale and may be complementary (e.g. friendly – unfriendly) or more gradable (e.g. human – machine) in their nature.

The results are a direct product of the participants' personal experiences, culture and beliefs, which means that participant selection and bias must be considered. To minimize this risk, a large and mixed selection of participants is advantageous [24].

4.4.3 Annotated Portfolios

Annotated portfolios are a tool to, in the combination between visual and textual material, showcase the decisions that went into the design. It is a way for designers to formulate their findings and point out why specific elements of their design are valuable, instead of leaving the artefacts speak for themselves, which often is not as obvious for the viewer, or user, as the designer might think. Annotations can touch upon many different subjects. Gaver and Bowers [39] list the following:

- functionality of the design
- the aesthetics
- practicalities in its production
- motivation for making the product
- the identities of the people, and
- sociopolitical concerns.

These points are guidelines for what aspects of the design to annotate and what motivates the designer's choices.

5

Process

This chapter outlines the process undertaken throughout the project. Adhering to the Design Innovation Process described in Section 3.2.1, our process follows a non-linear path, with various methods being revisited during the project’s course. The sections in this chapter are headlined with the methods employed, describing their execution and the findings that emerged.

5.1 Literature review

At the beginning of the project, a screening of relevant literature was conducted (Section 4.1.1) to gain knowledge about drones in general, usage of drones, control modalities and related work within search and rescue. As the review progressed, new areas of research surfaced and broadened the search for papers about cognitive load, human multi-control, and system transparency. Insights gained laid the groundwork for setting the project’s direction. The databases searched were *Google Scholar* and *Chalmers Library*. Words used for searching literature included *SAR*, *Search and Rescue*, *Drones*, *Multi-control*, and *UI*. In total, 29 papers were reviewed at different levels of detail. The most frequent types of sources used for the review were journal articles and different conference proceedings.

The main results from the literature review are presented in Sections 2.1, 2.3, and 3.3. Overall, the reading was a good way of familiarising ourselves with the topic. Some specific design insights, however, derived from reviewing related work. It is important to design for situational awareness and reduced cognitive load [15]. Systems can be of different automation levels where Ruff, Narayanan and Draper [16] suggests the level “management-by-consent” for this use case. Automation level ultimately leads to considerations about system transparency where Wang [17] points to the importance of finding a balance in the interface’s level of detail.

5.2 Meetings with the client

To understand the EOS-project more, the intentions, goals, and limitations, we had continuous meetings with the client during the thesis project. The meetings were held at SSRS’s office in Långedrag with SSRS’s head of innovation Fredrik Falkman.

The first meeting we had was to frame the project. Prior to the following meetings, we collected questions that had occurred during the process, such as “Will the crew have access to the footage from the rescue vessel?”.

As the project is made for a speculative future where regulations have been developed in a specific direction, many of the answers we received were “That is one possible solution”. This points at the ability for the project to develop further and that there are much more investigation to be done in the project that is outside of our scope. For example, the potential risks associated with this type of operation can not yet be determined, as it requires testing and real world experience. Nevertheless, Falkman indicated that the most significant potential risks are:

1. Hitting a person on ground/in water due to crash/uncontrolled landing.
2. Mid air collision with another aircraft.
3. Loss of drone.

The main results from the meetings are presented in Sections 1.3.2, 2.1.1, and 2.2 where the drone’s operation is described in more detail. To recall, the initial user of the interface will be someone within SSRS, preferably with a background in piloting or ATC. Similar to other volunteers, the user will be working on-call. In order to minimise the requirements of equipment that the user should have accessible, the interface should function optimally on a single monitor setup. This will make the work accessible from more locations, such as the home or the office.

Additionally, Falkman provided us with an operations manual (an internal document in progress) with the envisioned process of a mission, the parameters important for monitoring a flight, and the information that “the [drone operator] needs to determine from case to case based on the organisation’s collected experience if the potential benefit to a rescue mission outweighs the risk of [losing] the drone.”

Additional information retrieved from the operations manual is presented in the lists below.

The interface should display the following drone parameters:

1. Live position on map
2. Current and target altitude
3. Ground speed
4. Time to next waypoint
5. Time to target

6. Airspace information
7. No fly zones and geofences
8. Live positions and predictors of transponder-equipped low altitude aircraft, (ADS-b)
9. Live positions and predictors of transponder-equipped ships and SAR-aircraft (AIS)
10. Obstacles
11. Current weather
12. Live video

Further, the operations manual presented the following regarding how the drone is supposed to be handled:

1. Drone is allowed to fly between 0 and 45 metres above sea level.
2. Drone should avoid flying over people, boats/vessels and inhabited land/properties.
3. The standard mode of operation is Beyond Line of Sight (BLOS) for all parts of the operation; take-off, flight and landing.
4. Interface enable user modify battery endurance in the case of search mission
5. Provide location information of all drones

Lastly, the list below details technical specifications on the developed drone:

1. The drone is fixed-winged (mainly meaning it cannot hover in place, instead it circles (loiter).
2. The drone can be remotely identified via maritime AIS transponder service.
3. The drones maximum speed is 35m/s
4. When flying on maximum speed the drones battery can endure aprx 1 hour.
5. Battery endurance increase if drone throttle is decreased.
6. The drone is preferably launched against the wind

5.3 Scenarios

Scenarios (Section 4.2.3) were created in order to understand the drones' context of use. Based on information gained from SSRS (the drone's primary goal, drone technical specifications, what obstacles might occur, Swedish drone regulations etc.), seven plausible events were described in text frames, constituting the seven scenarios shown in Figure 5.1.

- Optimal Scenario,
- POI not found initially,
- Unexpected object nearby (While operating one drone),
- Incoming emergency call (Operating two drones in different locations),
- Two emergencies nearby, and
- Operating five drones.

All scenarios begin with *Pre-launch* and end with *Reached target*.

One of the scenarios describe the Optimal scenario, which was named after a scenario where no errors or obstacles occur during the flight. The optimal scenario starts off with pre-flight events, during which the drone operator receives an alarm from JRCC and accepts their request. The operator then receives the coordinates and subsequently begins the preparations for the flight by selecting the closest available and appropriate drone. The operator creates the route before initiating launch, by adding waypoints with position, altitude, and speed and ensures that the route is not crossing any restricted areas or static obstacles that might cause a collision. The system detects a potential risk with the route and alerts the user. The operator assesses the risk and makes a decision to adjust the route slightly according to the system's warning and launches the drone. The drone flies on autopilot according to the planned route to reach the target coordinates. The operator observes that all drone parameters looks good on the way and that no obstacles such as aircraft or ships are near. The drone reaches the target and starts loiter above the location while video recording. The operator selects a representative image of the situation and sends it to the rescue crew. They continue to monitor the location for any changes in the situation, and inform the crew of any updates. Once the rescue mission is complete, the crew inform the operator that the drone can land. The operator then lands the drone in the water near the location, and the rescue crew collects the drone, thereby concluding the drone mission.

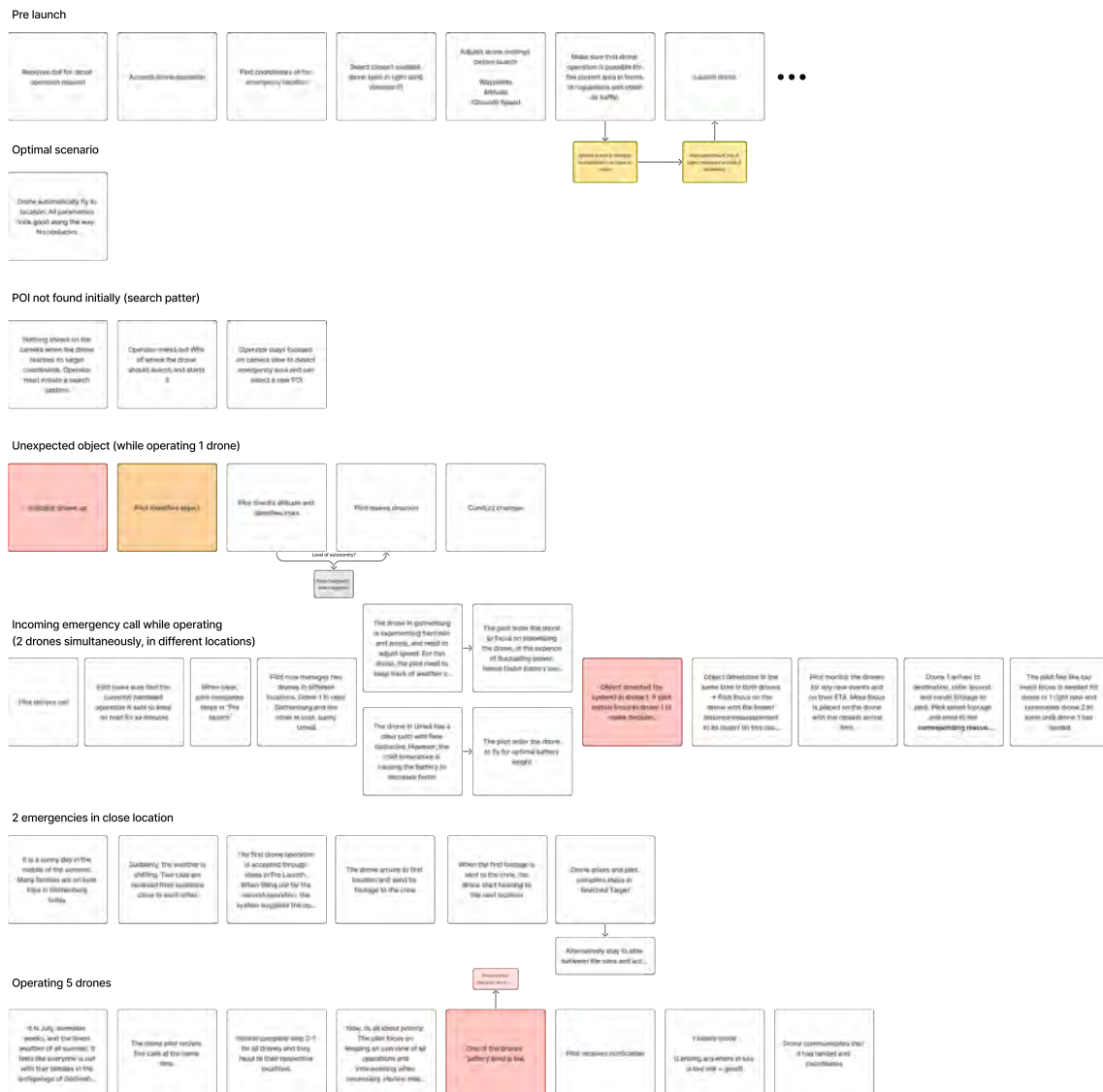


Figure 5.1: Scenarios covering seven different drone scenarios. Each scenario is captioned in bold text.

Initially the scenarios were used for guiding the benchmarking analysis, leading the way for what should be looked for in the different software. This set a basis for determining user flow and for identifying key frames prior to wireframing. Furthermore, the scenarios uncovered essential interface features that were to constitute an initial requirements table which can be found in Section 5.6. These features were then utilised to internally assess the early designs, ensuring that the design covered all seven scenarios. Ultimately, the scenarios laid the foundation for understanding the interfaces' requirements and mapping out the design space of the thesis.

Later on, the scenarios were employed in the workshop (Section 5.8) to provide participants with an understanding of the potential applications of the drone in various contexts. For enhanced clarity, the scenarios were paired with graphics and an-

notations during the workshop. Following the workshop, the scenarios were revised, incorporating insights from participants with experience of sea-rescuing routines and air-space operations. They informed us of eventualities the drone might experience, such as:

- Battery discharge,
- Nearby object on map or through camera,
- Connection is lost (C3 link),
- Risk of collision with boat or aircraft,
- Prohibited areas (populated land and restricted zones), and
- Drone flying to low for what is permitted.

This led to modifications of the scenarios, which subsequently informed the creation of wireframes, which will be presented in detail in Section 5.7. The final scenarios are presented in Figure 5.1.

5.4 Benchmarking

Benchmarking (Section 4.1.2) was done in order to screen the market of current advancements within the field of multi-control drone interfaces. Three GUIs were included in the benchmark and analysed in different formats due to differences in accessibility. Although there are other drone operating software (DOS) on the market, they require product licenses and are not publicly available. As a result, we chose the open-source system QGC, which can be downloaded and interacted with. The other two, UgCS and FlytBase, were not open-source but had available images and comprehensive reviews from which they could be analysed. As we were only able to interact with one of the GUIs, it is worth noting this as a limitation to be aware of when comparing the three.

QGC, is an open-source DOS, which was explored firsthand by downloading the software and noting how to plan a route, find information about drones and how to navigate within the map. By offering a wide range of customisable features, users can adapt it to be suitable for any flight with any type of drone. Additionally, QGC facilitates the capability to operate multiple drones simultaneously.

UgCS is a professional DOS. It was experienced and examined through a video streamed webinar. Similar to QGC, UgCS has a broad area of use to suit any flight. To operate multiple drones two software are required: *Enterprise* and *Commander*. The user in the video employs two computer screens to enhance their overview of the two software applications. It is unclear whether the same comprehensive view could be achieved by presenting both applications side by side on a single screen.

FlytBase is a DOS for monitoring drones in warehouses, at construction sites, or for public security. The product is marketed as “Built for BVLOS Drone Operations” and promotional materials feature mock-ups of multi-drone operating stations with up to seven screens. Some reviews, however, indicate that a single-screen use works fine for specific purposes.

A table was created to list various features for identifying design patterns (Appendix B). By colour-coding cells in green or red, information about which software included specific features was organised. Features partially implemented in the software were marked in yellow. By going through the user flow led by scenarios from the storyboard, both essential, and desirable features absent in all three interfaces could be listed. The GUIs shared three key elements: a map, a camera view with a video stream, and the capability to set waypoints in the map through a click-and-drag function (i.e. user selecting where in the route the drone will make changes in altitude, speed, direction, etc. and moving waypoints in the map using the cursor). It became clear that the compared DOSs are designed for diverse applications, types of drones and contexts, offering detailed manual flight preparation. This manual approach is beneficial as it covers multiple use cases. However, as Cooper, Reimann, Cronin *et al.* [40] explains, excessive customisability within an interface can hinder navigation. Instead, by keeping to only necessary controls, user excise is reduced and orientation enhanced. The use case of SSRS drone is of a more static nature, with no need to configure different drones etc. Therefore, instead of spending time on specifying the context for every mission, the user should rather be encouraged to spend less time on preparing a flight, to be able to launch the drone early for the crew to receive footage on time.

5.5 Semi-structured Interviews and Observations

Interviews (Section 4.1.3) were conducted with the aim of gaining insights into how established fields utilise their interfaces and how their routines work regarding multi-operations. For this purpose, we contacted a drone pilot, several drone platform companies, Säve ATC, Landvetter ATC and an autonomous truck company. Out of which, we were able to hold interviews with the three participants listed below. All interviews followed a semi-structured approach where questions were altered depending on the context. The set of questions included “What are your preparations before a drone flight and what parts do you prepare using the DOS?” (specifically asked to a drone operator), “How many and what physical components are you interacting with and can any of these be handled digitally instead?”, and “What level of autonomy does the system provide? When do you need to operate things manually?”.

During the interview, follow-up questions were asked to make it more conversational and to get more into detail about information that could be useful for designing a drone interface. The complete interview template is available in Appendix C.

Employed at Aviant, **Marcus** is project lead and drone operator for a project in

collaboration with Aviant and the region of Västra Götaland (VGR), developing a drone service for medical goods delivery. The drone is intended to transport medical goods between hospitals and healthcare centers in the region. Marcus operates the drone BVLOS and was contacted for the purpose of retrieving insights about preparations, planning, safety precautions and flights when operating drones BVLOS.

Rickard works at Landvetter air traffic control (ATC) as the ATC manager. An air traffic controller's responsibility is to ensure that aircraft taking off and landing at the airport do so safely. They need to keep track of several aircraft simultaneously and therefore possess good multitasking skills. The qualities of an air traffic controller are sought after for SSRS's drone project, because the goal of SSRS's drone interface is to enable multiple drone flights, which is why they are a part of the prospective user group.

Magnus monitors Einride's autonomous, unmanned trucks when they are on the road. We contacted him in order to broaden our perspective and gain insights beyond our immediate scope. This approach aimed to prevent narrow assumptions based solely on other drone interface setups, enabling us to explore alternative remote operation solutions.

In connection with the interviews, we were able to observe the participants in some way (Section 4.1.4). We were invited to Aviant's test flight with VGR and were able to follow Marcus' actions from preparations and take off to landing in first hand. Rickard presented an air traffic controller's setup, both physical components as well as their digital interface, similar to how Magnus showed us a demo of how Einride's monitoring setup looked like.

After interviews, an affinity diagram was created collecting valuable quotes, and notes from the three meetings. The collective insights were narrowed down to align with the thesis objectives and then divided into five categories that are described in the following subsections.

5.5.1 Operational context

All interviewees work with a digital setup. Einride uses three large monitors at different angles around the operator. Magnus mentions, however, that three screens might be excessive and that they are considering reducing this to one monitor with different views depending on the scenario. The only physical artefacts are an emergency break and a small keypad for quick commands. We observed that ATC at Landvetter uses multiple displays, radios and physical flight strips. However, Rickard was opposed to the use of physical flight strips because they occupy one of the operators' hands, which can be seen as a situational disability, as they cannot navigate freely with the rest of the equipment.

“När det är stripphantering så skriver vi med pennan i ena handen och har radarn i den andra handen, och har du headset då har du en

sändknapp ... så man blir verkligen handikappad för man behöver de här händerna.” [When handling the flight strips we hold the pen in one hand and the radar in the other, and also the headset with a send button ... so you really get handicapped because you need these hands.]

When visiting the test flight, we observed that Marcus from Aviant uses a tablet to control the drone. During the visit, it became clear that the tablet is easy to navigate and also provides a lot of auditory feedback. It is important to remember that Einride and Aviant work with a one-to-one control relationship, for which their equipment is optimised. ATCs operate multiple aircraft simultaneously, which is very relevant to the aims of this thesis. Furthermore, in recent years, remote ATC towers have emerged, a role that Rickard suggests is even more closely aligned with remote drone operators. However, while they can operate from a distance, they are currently not permitted to manage multiple airports simultaneously, something that is believed to be a future possibility as the profession is moving towards increased automation.

“Det kommer vara automatiserat. Det kommer vara massa hjälp och stöd ... vi kommer gå mer åt övervakning.” [It will be automated. There will be lots of help and support ... we will head more towards monitoring]

5.5.2 Regulations

The introduction of the EOS project naturally raised questions and concerns about regulations among the interviewees, who are well-versed in current airspace regulations. SSRS’s drones are set to operate across all of Sweden, indicating that their flight paths will often intersect with shared airspace, known as D-areas. Rickard emphasises the critical need within these areas to maintain visibility of all other airspace users to ensure safe separation. He highlights the use of transponder radar in the ATC interface as a vital tool for monitoring other aircraft, a feature he believes is equally essential for drone operators. Even though the drones will fly at a significantly lower altitude than most airplanes, there are private actors and military aircrafts that operate at the same altitude and will need to be avoided.

“... sannolikheten är nästintill obefintlig, men om du kör in den där lilla Cessnans [militärflyg] enda motor med även din lilla drönare så kan det orsaka att den kraschar med 3 personer ombord.” [... the probability is almost non-existent, but if you ram that little military aircraft’s single engine with even your little drone, it could cause it to crash with 3 people on board.]

In addition to aircraft, Marcus from Aviant underscores the necessity for drones to avoid flying over boats, utilizing Automatic Identification System-maps (AIS-maps) for ships. Ultimately, it was evident from all interviewees that keeping to regulations and ensuring safety is paramount, and the operating system must be able to handle any eventuality, regardless of its improbability.

5.5.3 System

Similar to an ATC, Marcus at Aviant “commands” the aircraft to take certain actions. These actions are pre-planned by Marcus himself and the drone flies accordingly. The route is mapped out using a tablet with an open-source drone operating software connected to the drone. If the drone for some reason should deviate from its route, it also has predetermined safety precautions and enters a fail-safe mode to cancel its route and land in the closest of the many marked safe coordinates. Marcus, unlike Rickard at Landvetter, operates only one vehicle but believes that it is possible to monitor more with an adapted software:

“Man måste göra en plattform som har ett bra ändamål för det – att det faktiskt fungerar. Individuell information, men samtidigt samlad information.” [The platform needs to be adapted for its purpose, in order for it to work. Individual information, and aggregated information at the same time.]

Related to this, Magnus at Einride also points out the importance of not presenting too much information to the user as this might overwhelm them, which could lead to increased stress and decreased control. It was clear during observation that Einride has worked on reducing excessive information.

The level of autonomy was discussed in the interviews. Rickard thinks that “automation is the key to multi-control of drones”. Both literature and Marcus point out that management-by-consent would be the preferred option, which means that the system should detect obstacles, make suggestions and inform the user which in turn makes the final decision to accept the suggestion or to change it. It is important to not make the system too automated, as automation leads to decreased focus. Marcus points out that when the focus is low, the operator becomes inactive and “laid back”, which according to him is not good because that leads to slow reaction, if the operator needs to take action in an emergency.

To fully trust automatic or partly automatic systems it is important to inform the user that it works well according to Rickard. Landvetter ATC uses several systems as backup if one fails, similar to SSRS including a redundant GPS-tracker in their drone. This is important to SSRS as a confirmation that their system works as intended, which increases the reliability of the system and the user’s trust.

5.5.4 Interface

The importance of a good overview has been expressed by all interviewees, which led us to prioritise the issue. Creation of a good overview includes carefully selecting what, how, and where information should be presented. The interviews informed the project about what parameters should be considered high and low priority information.



Figure 5.2: Photograph of the flight monitor used at Landvetter approach control

An ATC's main duty is to prevent potential collisions between aircraft. If such risks are identified, controllers instruct pilots to ascend or descend, a process known as *separating the flights*. If (in worst case) all systems are down, the controller needs to separate visually, meaning that they physically observe aircraft from the windows of the ATC tower to guarantee a safe distance between them. This safety procedure motivates the inclusion of camera view in a drone interface, especially because not all vehicles the drone needs to separate from are visible in the transponder- or AIS map. The camera view would also work as a redundant position validator, as it can show the user's expected landmarks in a video feed.

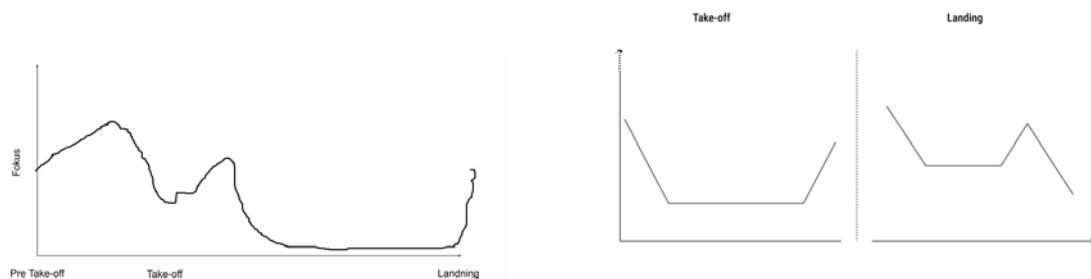
Furthermore, the ATC works with radar systems, which allows them to see aircraft nearby digitally on a simple map (Figure 5.2). Because of the airplane's altitude, the map they work with does not require much detail. Rickard therefore recommends including a more detailed map for a drone operator, because drones fly at a much lower altitude and the risk of obstacles increases, as well as constant altitude information. Marcus however claims that altitude is more or less irrelevant for him (when the drone flies as intended) because he always flies over water and has therefore no ground level to consider.

Additionally, as mentioned earlier the level of autonomy is considered a crucial aspect to consider. With a balanced level of autonomy, the system should be able to communicate warnings to the user, if for example flight predictors, ships or other obstacles on the route are detected. Warnings should be presented clearly in order for them to be easily identified by the user.

5.5.5 Stressors and Human Factors

All interviewees are highly experienced and have received extensive training in their respective fields. Marcus at Aviant is a former pilot and has expressed that he is comfortable in his current role as a drone pilot. Magnus at Einride is a former truck driver with the experience needed to now operate an autonomous truck, and naturally, Rickard has had many years of education as an ATC. These professions depend on human abilities and demand a high cognitive ability to ensure efficiency and safety. Similarly, a drone operation will depend significantly on the operator's ability to act and make decisions. Both Magnus and Rickard emphasise the importance of breaks in order to maintain a high level of focus when working. They also point to the fact that they never work alone, although Magnus refers to a remote college. The knowledge that someone is there to assist them is a comforting factor. Rickard notes that within ATC, operators become accustomed to stress resistance, yet it should be avoided. He proposes that maintaining structure within the interface, minimising external disturbances and increasing automation within the system are concrete ways to reduce stress. In addition, Marcus posits that there is a correlation between stress and focus. He asserts that while excessive manual work can lead to stress, excessive automation can lead to a lack of focus.

“Stress är negativt vinklat skulle jag säga. Stress är ju också en sak som är positiv därför att det höjer vårt fokus.” [Stress is negatively charged, I would say. Stress is also a positive thing because it increases our focus.]



(a) Coordinate system in which Marcus at Aviant has drawn his level of focus during a flight, including pre take-off, take-off, and landing.

(b) Coordinate system in which Rickard at Landvetter has drawn his level of focus during take-off, and landing at Landvetter, from start to finish.

Figure 5.3: Perceived focus levels for monitoring flights in different ways.

Marcus illustrated his focus levels during a flight through a chart (Figure 5.3a), indicating that most focus is required before launch, during the launch and when landing the drone. Similarly, Rickard illustrated that the take-off and landing of a plane require the most of his focus (Figure 5.3b). Rickard explains that his control and focus decreases in the end of the landing, when the pilot takes over the responsibility. In both graphs, it peaks at the beginning of the process, when control is being established, and at the end, when the actual take-off or landing occurs, given

that it is the phase of flight with the lowest altitudes and highest collision risks. Although the graphs are not directly comparable, as Marcus has illustrated his focus level of a full flight, and Rickard has isolated it to the take-off and landing, there are some similarities to highlight. There are sequences in both graphs where the focus is low, something Marcus points out might affect reaction speed.

Moreover, none of the interviewees had encountered any major hazards throughout their careers, fortunately. It was evident that having trust in the systems they operate is essential, a trust that stems from the reliability of the technology. Additionally, Marcus mentioned that, as a former pilot, he had essentially been trained to trust in the systems he operated, which both served as a stress reliever and improved his performance. ATCs are trained to manage multiple flights and ensure overall safety. However, it is the pilot who holds the ultimate responsibility for the aircraft. In the role of a drone operator, Rickard presumes that the operator will carry the responsibility of both ATC and pilot. This dual role could be pressuring and underscores the importance of staying updated on regulations to prevent both internal stress and external accidents. In this scenario, Rickard proposes that increased system support could alleviate the need for operators to retain regulatory information mentally. He adds that ATC regulations are increasingly being integrated into their systems for the same reason.

5.6 Table of Requirements

During the process, insights have been collected and formulated in a table of requirements. A first draft was established through insights from the Literature review, Scenarios, Benchmarking, Interviews and discussions with the client. The table contained requirements both directly and indirectly affecting the interface. This version was later used to guide the creation of first draft fireframes. After receiving feedback from participants in the workshop as well as internal feedback from the PUEA, the requirements table was revised. The second draft discarded requirements that appeared irrelevant to the scope of this project. The updated requirements were ranked by prioritisation: *must*, *should* and *could*. The final draft is presented below in Table 5.1.

Requirement	Priority
UI Elements	
Easy access to highly prioritised information	Should have
Visually clear that track is free	Should have
Nudge the use of map layers	Could have
Nudge the user to constantly check values during flight (camera, map, drone parameters)	Could have
Warn when drone is not flying on maximum altitude	Could have
Operation	
Multi-visualisation of maps	Must have
Ability to pause drones	Must have
Switch between camera and map view within 0-1 action access	Must have
Information Overview	
Offer overview of prioritised information	Must have
Offer detailed information on request	Must have
Hide irrelevant information (carefully prioritised)	Could have
Parameters (system transparency)	
Drone parameters (altitude, direction, speed, climb rate, throttle)	Must have
System informing about fail-safe when engaged	Must have
Estimated time of arrival to target location	Should have
Information about next WP	Should have
Time and distance difference between direct and modified route	Could have
Ability to fetch coordinates	Could have
Map (Information Visualisation)	
Visualise topography	Must have
See other vehicles (transponder)	Must have
Visualise drone altitude	Should have
Determine obstacle heights	Should have
Visual NOTAM	Could have
Other	
Drone operator undergo pre-training	Must have
Max 2 actions to access NOTAM	Should have
Receive coordinates by text	Should have

Table 5.1: Table of requirements divided into six categories. Each requirement is ranked in priority from could, should and must have.

5.7 Wireframes & Wireflow

During our first visit to the synthesis phase, wireframes (Section 4.3.5) were created in Figma. User flow diagrams (Section 4.3.4) can be created prior to wireframes in order to ensure that the users complete specific tasks efficiently in the interface. However, for our interface the main goal is to facilitate the monitoring of a full frame, rather than to optimise the user flow, which is why we decided to opt out user flow diagrams. Instead, we started directly with creating wireframes with the aim to iterate these after recurring evaluations. To spark ideation, we did a variant on brainwriting (Section 4.3.1), where we began by individually sketching out ideas on an interface that would cover the seven scenarios generated in Section 5.3. Then, we switched to work on each other's concepts, resulting in concepts 1a and 1b presented in Figure 5.4. At this stage, focus was on functionality and when the two wireframe concepts were created they were evaluated internally by being compared to the table of requirements which resulted in a combination where design patterns from each concept were selected and merged into one elaborated version (Figure 5.5).

Concept 1a



Concept 1b

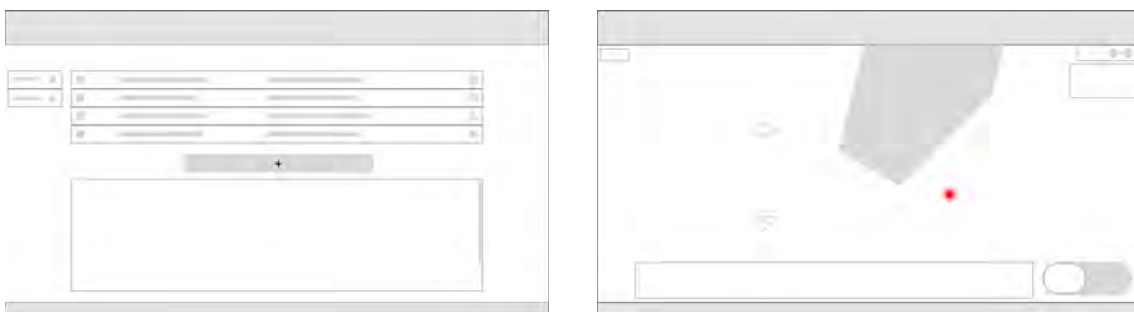


Figure 5.4: Four frames of wireframe concepts 1a (top frames) and 1b (bottom frames), developed through brainwriting using Figma. The right frames illustrate each concept's layout for overviewing multiple drones. The left frames illustrate a full screen view of selecting a drone.

Concept 1 (a+b)

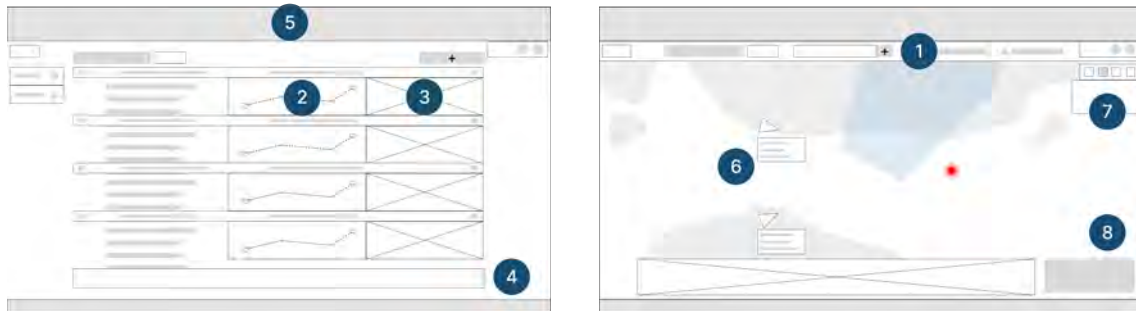


Figure 5.5: Two frames of the first wireframe concept that emerged by combining features from concept 1a and 1b (Figure 5.4). Numbers in the image mark particular elements and are referred to in the text.

Since it is important that the drone launch quickly, We appointed the coordinate finding primary component and decided therefore to place it in a fixed navigation top bar for a one-click reach at all times ① (Figure 5.5). Included in the wireframes are placeholders for map ②, camera view ③, Notice to Airmen (NOTAM, aeronautical information) ④, launched drones ⑤, information about drones at stations ⑥, local weather forecast ⑦ and a launch drone button ⑧. The user has the ability to type in coordinates received from the emergency call, select a suitable drone, add and adjust waypoints, and launch the drone. The main objective of the interface is to allow for multi-operation of drones, while still encompassing crucial functionality such as finding the emergency position, launching an appropriate drone, and avoiding obstacles on the route – both static objects such as islands, wind turbines etc, and dynamic objects, such as ships or aircraft. While a definite limit was not yet decided, the initial draft was largely confined to four drones to what seemed convenient. The layout was tested and adapted to the different drone quantities, displayed in Figure 5.6 and Figure 5.7, where the wireframe concept allows for two monitoring modes. A full-screen mode with up to four drones (Figure 5.6), or a drone-list mode with scrollable and collapsible vertical cards (Figure 5.7). The idea was to enable different detail levels for different quantities or to the specific user’s preference for an adaptable layout. The wireframe concept was later evaluated with the HTA- and PUEA-methods (Section 5.9). This was followed by mapping out a wireflow, to clearly describe the interactions with the components for the workshop participants (Section 5.8), where the concept was used as a way of mediating the scenarios.

A second draft of wireframes was created after conducting the PUEA as well as receiving feedback from workshop participants. For example, in the first draft, the drone information is only visible when hovering on the drone as a mean to minimise excessive information. However, the PUEA highlighted a possible error being 3.2.1c in Appendix D: “User does not find drone stats when determining drone selection”, which can be prevented with “Information always visible”. Hence, in the new wireframes the drone information is static instead of being hidden behind a hovering state. Another possible error highlighted from the PUEA is 4.1c: “User

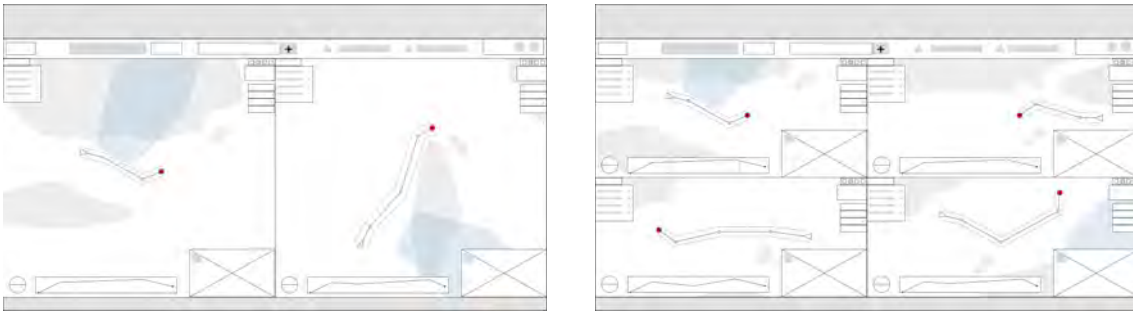


Figure 5.6: First draft wireframes splitting the screen when monitoring a) two drones and b) four drones.

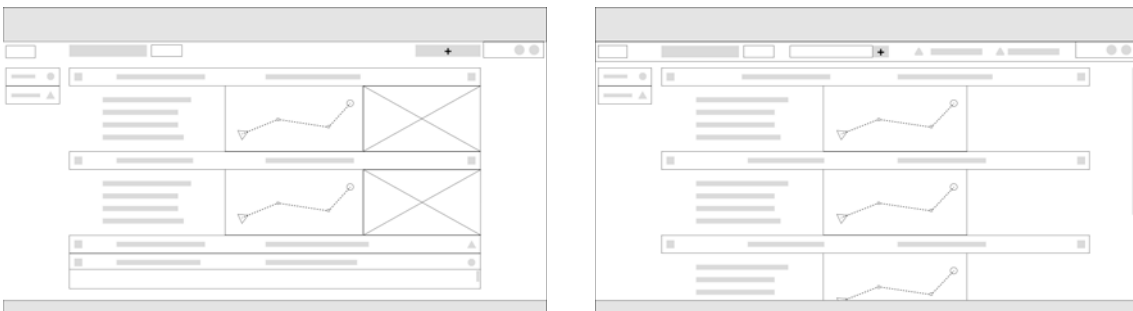


Figure 5.7: First draft wireframes of the overview layout when monitoring four drones. a) Two drone cards are collapsed. b) No drone cards are collapsed creating a scrollable frame.

launch drone before finished route plan”, where the prevention solution was to add a slide-action to the launch button, to prevent the user from accidentally clicking on it (1) (5.8).

The workshop participants identified specific needs that were incorporated into the wireframes. The participants believe that presenting information side by side is an effective approach. However, they felt that the current structure was overwhelming, so it was redesigned to divide and prioritize essential information at different detail levels. The idea was to implement tabs in which the individual drone information in the overview would be aligned horizontally (Figure 5.8, rather than vertically, as it was in the previous draft. This enables the tab bar to serve as both navigation and a means of maintaining the most essential information in view at all times. As a consequence of the implemented tab bar, the navigation top bar from the first draft was reworked into a navigation rail to the left (2). Furthermore, participants expressed a desire for the ability to pause drones in order to manage multiple simultaneously, where a loiter-button (3) and land-button (4) was added as a quick action. Continuing, it was important to immediately tell if the drone is descending and that the path ahead is clear. Hence, a placeholder for altitude schematics was added (5), drawing inspiration from the component used in the software QGC examined in 5.4. Lastly, participants provided insight into an essential aspect of maritime rescue operations, namely the ability for the drone to search the area upon arrival. This implies that in certain instances, the drone may be required to remain at the scene

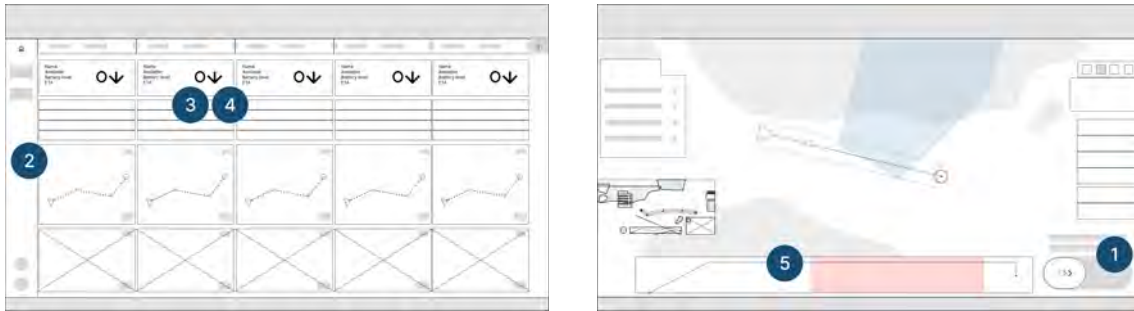


Figure 5.8: Second draft of wireframes, with numbers referring to updated features described in the text.



Figure 5.9: Feature for determining the optimal time at scene in combination with time to the target. The border radius of the circle represents how far the drone can travel with the current configuration of ETA and ETS.

for a longer duration rather than arriving as rapidly as possible. Hence, we worked on a pattern for visually configuring this matter in the interface. The implementation is seen in Figure 5.9 which illustrates a circle with a radius that is dependent on the fields Estimated Time of Arrival (ETA) and Estimated Time at Scene (ETS) and represents how far the drone will be able to reach depending on those variables.

5.8 Workshop

A workshop was conducted with members of the intended user group to gather insights on their needs for an interface to control drones during a sea rescue mission. The workshop aimed to keep user needs at the forefront of the design process and to guide the future direction of the design, following the co-design methodology described in Section 4.3.2. Additionally, the workshop was intended to receive feedback on the benchmarked interfaces, as well as to evaluate the scenarios and wireframes. It was also hoped that the workshop would generate ideas for possible design patterns created by participants as a way of involving users in the design process. Initially, we considered mapping out the user group through User Profiles (Section 4.2.4). However, due to the low sample size and the fact that the workshop participants did not ultimately represent the intended user group, it was disregarded as irrelevant.

To recall, the user group was defined as volunteers within SSRS with a flight-related background. To ensure a close to accurate representation of the user group, all recruited participants are or have been pilots, air traffic controllers, drone operators, or similar and are or have been volunteers at SSRS rescuing stations. Table 5.2 shows that a total of eight participants took part. The workshop was conducted remotely over Microsoft Teams, with the use of FigJam (“A visual collaborative whiteboard where teams can diagram, brainstorm, and organize ideas together” [41]), for presenting and sharing thoughts. The entire workshop was video and audio recorded for reviewing and transcribing purposes, to which all participants agreed through a consent form found in Appendix E.

The workshop lasted for three hours and the agenda is listed below:

- Introduction
- Delegate presentation
- Benchmarking discussion
- Scenario walkthrough
- Break
- Individual comments
- Design
- Reflections on the session

Background	SSRS role	Rescue station
Air Force Pilot	Student	RS Kåringön
Flight Technician	Teamleader	RS Munsö/Ekerö
Private Pilot	Skipper	RS Dalarö
Helicopter Pilot	N/A	RS Rörö
Commercial Pilot	N/A	RS Uppsala
Drone Pilot	Skipper	RS Ystad
Private Pilot	N/A	RS Munsö/Ekerö
Commercial Pilot	Skipper	RS Stockholm

Table 5.2: Workshop participants

The workshop began with a presentation of ourselves, our thesis project and how it contributes to the EOS project, and the purpose of the workshop. To familiarise

everyone with each other, all participants were asked to introduce themselves, their professional and SSRS backgrounds and if they wanted to share any previous experience on the topic. This was an effective method to engage everyone and establish a comfortable setting, particularly in the context of a remote meeting.

Continuing, we presented the three drone interfaces that were examined in the benchmarking and allowed for comments on their appearance, perceived functionality, and overall impression. The participants unanimously agreed that none of the interfaces were very good, describing them as “confusing” and “poor overview”. It is important to note that since the participants could not interact with the interfaces, the results might hold weaker credibility. Overall, this section of the workshop did not deliver as intricate results as expected, but was a useful way of warming up participants to the idea of what a drone interface might look like.

To deepen participants’ understanding of the project’s objectives, we presented three example scenarios in the form of scenarios, demonstrating how different situations could be handled by a drone interface. Wireframes, accompanied by annotations, were used to illustrate workflow and functionalities, where efforts were made to maintain a balanced fidelity level to avoid limiting participants’ perspectives while ensuring clarity (see Figure 5.10). Participants were asked to comment on each step in the scenarios and write notes of suggestions, which naturally led to discussions about preferred approaches in different situations. The group unanimously agreed that operating multiple drones was a flawed idea to begin with, with comments such as

“har du väl startat ett uppdrag ser jag det som omöjligt att ta ansvar för fler än en drönare.” [If you are in operation of one drone I find it impossible to take responsibility for additional drones],

and

“att hålla koll på alla dessa sensorer samtidigt blir information overload” [keeping track of all of these sensors will lead to information overload].

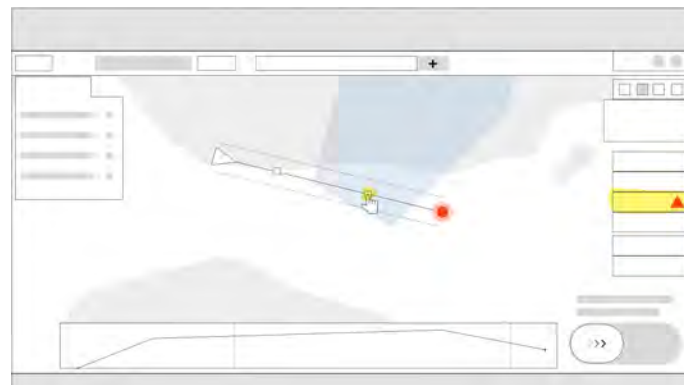


Figure 5.10: One example of the wireframes presented at the workshop, enhanced with highlighted areas, colours, and cursor.

Ultimately, the act of multi-operation seemed too demanding according to participants, pointing to air-space safety, multi-tasking difficulties, personal responsibility and information overload. In addition to providing feedback on the wireframes, new insights emerged about the rescue team's working methods, laying the foundation for framing new user needs and topics for further investigation. One participant explained that

“Oftast så hittar man ju inte objektet direkt utan det kommer löpande information” [Usually, the object is not found immediately, but require additional information].

Given that the target position is not always accurately reported and that it may change over time, new insights emerged that require consideration for the interface design.

The final activity of the workshop was a collaborative design session with the aim of generating ideas about the appearance and functionality of the interface, as well as an approach to uncover more user needs. Participants were presented with the scenario *What could an interface look like, and what should it contain when piloting three drones at the same time?* and were tasked with creating a concept together through discussion and ideation. To establish a basis and to guide the design process for non-designers in the group, we provided a set of components, similar to those found in wireframes, that participants could drag into a computer screen mockup. Despite our efforts to engage, this task proved to be challenging as participants appeared hesitant to take action. This reluctance may have been due to the on-line workshop format, which potentially hindered effective collaboration, along with participants' unfamiliarity with FigJam. Furthermore, the negative view on multi-control uncovered previously in the workshop may have affected the group's devotion to design for this specific scenario. Despite not meeting our initial expectations for a visual design concept, the session was valuable as it led to the development of a requirements table created by the participants. This table highlights the essential user needs and other vital functionalities that they believe the interface must have:

- Side by side map views (suggestion of tabs),
- Able to pause drones,
- Display of hinders ahead,
- Drone climb/decline information,
- Drone speed,
- Direction, and
- High automation but being able to take manual control when necessary.

Overall, the workshop provided valuable insights into many aspects of the project. While certain anticipated outcomes were not attained, unexpected insights emerged. The organization and planning of the workshop were successful, leading to active participation, engagement, fruitful discussions, and adherence to the schedule. However, we found the online format to be less preferable, and we believe that a real-life meeting would have been more valuable as a workshop format. The limited use of online tools may have hindered creativity and collaboration among participants. Key insights from the workshop are highlighted below.

One of the main insights was that participants appeared sceptical towards the feasibility of operating multiple drones simultaneously, based on their comments during the workshop (some of which have been cited in this section). Participants have observed the development of drone activity over recent years and have seen the practical and regulatory challenges of operating a single drone beyond visual line of sight (BVLOS), including safety risks and the complexities of coordinating with other airspace stakeholders. Given the challenges that currently exist in operating a single drone, the notion of one pilot managing multiple drones is considered a leap too far.

“Flygsäkerhet är ett krav och det faller om man har flera drönare i luften samtidigt” [Air-space safety is a requirement and it can’t be fulfilled if you have several drones in the air at the same time].

Insights were gained regarding the workflow of sea rescuers. These findings will be used to refine the hypothetical scenarios to better reflect real-world situations, allowing us to address the actual needs of users when designing the interface. Participants emphasized the unique nature of each rescue mission and expressed a desire for the ability to manually control the drone when necessary. However, the workshop revealed a strong counter recommendation to manual control, but possibilities for enhancing the level of control will be explored based on their feedback.

Further, the workshop uncovered the relative importance of interface features, helping to prioritise components within the interface. Although desires and needs were mostly identified in general terms, participants also specified significant design patterns that they preferred over others, giving us direction as to what should be included in the interface.

Upon analysing the workshop results, we synthesised them into a list of considerations that could be relevant to the EOS project. To adhere to the aim of this thesis, the list was narrowed down to relevant considerations and translated into specific requirements for the table of requirements.

5.9 Predictive Use Error Analysis (PUEA)

To evaluate the first draft of wireframes, a PUEA (Section 4.3.8) was conducted (Appendix D). Prior to that, a hierarchical task analysis (HTA) was employed to

inform the PUEA. A combination of the two methods was conducted in order to reflect on the wireframes and collect insights on what could or should be changed for future drafts.

The HTA was conducted internally. The main goal of each drone flight was defined as *crew receiving footage from the emergency situation*. The task was divided into seven subtasks:

1. Prepare for on-call (drone operator)
2. Accept drone operation
3. Plan drone route
4. Launch drone
5. Operate drone flight
6. Send footage
7. Finish flight

which in turn were divided into subtasks. The highest levelled subtasks were defined as operations.

Following the creation of a large HTA tree was the predictive use error analysis (PUEA). Sticky notes were attached to each operation and subtask. Errors which might occur for each operation, or task, were written down in accordance with PUEA template:

Level 1: Analysis of functions

1. What happens if the user performs an incomplete operation or omits an operation?
2. What happens if the user makes an error in the sequence of operations?
3. What happens if the user performs functions/tasks correctly at the wrong time?

Level 2: Analysis of operations

4. What can the user do incorrectly in this operation?
5. What happens if the user performs this operation at the wrong time?

Tasks and operations that were not directly related to the wireframes were neglected.

A PUEA matrix was created with the PUEA sticky notes as a basis. The errors were categorised into different types and causes. *Error consequences* were ranked by severity on a scale of 1 (most severe) to 5 (least severe). While probability wasn't explicitly factored into the analysis, following Bligård and Osvalder [38], events that were too unlikely to occur were disregarded. *Detection probability* was also estimated on a scale of 1 (least likely) to 5 (most likely). The results of this analysis, presented in four separate matrices in Appendix F, mapped – Consequence versus Task, Error type versus Task, Error cause versus Task and Detection probability versus Task. Colour coding within these matrices highlights critical moments in the tasks where errors are most likely to occur. A summary accompanying the matrices (presented below) further clarifies these key findings.

Consequence versus task (shows in which tasks the most serious consequences of errors exist):

- Most severe consequences occur when the route is wrongly planned
- Severe consequences occur when the drone is wrongly monitored
- Mild consequences occur if the drone has failed to launch or if it lands in an inappropriate area

Error type versus task (shows which type of use error exists in the various tasks):

- Most common error type is *Planning* (refers to planning actions to execute a task)
- The operator might miss checkings during flight
- During planning of a route, the most common error type is *Planning* (P1: Plan preconditions ignored or P3: Correct but inappropriate plan executed)
- During a flight, the most common error type is *Checking* (C2: Checking incomplete or C4: Wrong check on the right object).

It is important not to confuse the error type *Planning* with planning of the drone route. The error type refers to errors in planning actions to execute a task, in this case the task is planning of the route

Error cause versus task (shows the causes of the use errors in the different tasks):

- The most common error cause is *Knowledge based*
- A common error type is *Slip*, failure of attention during execution
- Pre-training is essential for the drone operator to learn how to adjust waypo-

ints, what rules and laws that must be followed and why

- Pre-training is essential for the drone operator to learn how to monitor the flight and what to look for in the system
- The system should notify the drone operator of any deviations

Detection versus task (shows in which tasks there are errors that are difficult to detect):

- It is important to detect errors during the planning of a route, or it will affect the safety and/or efficiency of the mission, alternatively the drone will not launch, and the mission will be cancelled
- During the flight, the pilot might detect errors if they are attentive. This requires a large amount of focus, the system should help the user to recognise errors by being transparent
- It is easy for the user to detect errors both during launch and landing

For each detected error in the PUEA matrix, an approach for preventing it was formulated. These preventions will be taken into account for updating the table of requirements (Section 5.6) for creating a second draft of wireframes. The four result matrices combined uncovered the nature of error types in the system, leading to insights on where the design needs improvement for preventing errors and what errors lies outside the scope of this project. Some aspects may not be directly affected by the interface and are more likely to be effectively prevented through other parts of the EOS project. For example, error 3.3.1b “User miss important information due to not using map-layers” in Appendix D can be prevented by “Nudging the use of map layers” in the interface. Error 7.2 “User lands drone on inappropriate spot” could have been prevented by including an information manual in the interface. However, with intentions of keeping the user’s focus on the primary goal and minimise excessive information, it was considered to be more effectively prevented through pre-training where the user get situational training and knowledge on all regulations. The suggested prevention method was therefore “Well defined required distance [between human and drone], drone flying course”. While of importance, this fall outside of this project scope.

5.10 Prototyping

Previous steps in the process have generated insights that have defined requirements and considerations for a drone piloting interface. The following phase, prototyping (Section 4.3.6), aims to concertise these insights. The prototyping was conducted using Figma, with the objective of refining the latest wireframe iteration to produce a pixel-perfect interactive mockup. Focus was on covering the seven scenarios for the upcoming user evaluation. An interactive mockup will be a means of evaluating



Figure 5.11: An overview of the component library along with building blocks for the interface.

both the design and the design considerations. Furthermore, it represents a valuable deliverable for the client, serving as a foundation for development.

After reviewing the wireframes, we proceeded to refine the components for the high-fidelity prototype. As the wireframes provided the foundation for the interface, the initial step involved establishing a fine-tuned component library (Figure 5.11) applicable across all frames. This was done in accordance with fundamental guidelines from Google’s Material Design [42], ensuring component measurements, padding, and margins aligned with a 4-pixel grid, with the objective to ease future development of the interface. This phase also contained brainstorming for solutions to specific targets expressed in the table of requirements. We considered using morphological matrix (Section 4.3.3) for these more specific tasks. However, it was deemed more comprehensive to design several options and validate them against each other, rather than breaking them down into subtargets. For example, a functionality required was the ability to set the speed of the drone in relation to the time it should endure at the locations. After developing several low fidelity concepts for this target, it was refined to one component as seen in Figure 5.12. Once key frames were established, we shifted focus towards enhancing the prototype’s interactivity. Interactions between frames define the user flow, and as previously stated, we prioritised covering the established scenarios. Further detail on the prototype results can be found in Section 6.2. This phase culminated in an interactive prototype prepared for user testing which is described in Section 5.12.

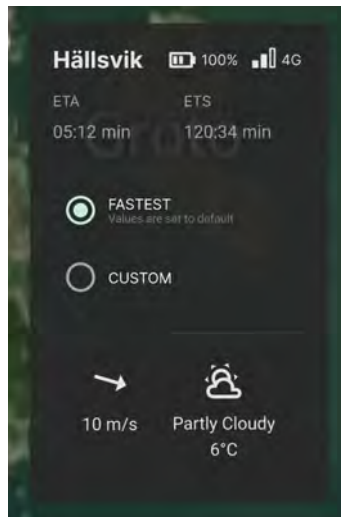


Figure 5.12: The card in this figure shows how the user can change the drone speed from the fastest to a custom speed with radio buttons, to enable greater endurance for the drone.

Launch	00:00 min
Take off	00:00 min
WP	00:00 min
WP	00:10 min
POI	02:10 min

Figure 5.13: List of waypoints during a flight. The WP currently headed to is highlighted in white.

5.11 Heuristic Evaluation

The design was initially evaluated through heuristic evaluation (Section 4.3.7) in its early wireframe-stage. The ten heuristics have been revisited continuously during the design process in order to ensure that the interface met baseline UI guidelines. The following paragraphs describe how the heuristics have been covered in the final prototype.

1. Visibility of System Status. These factors were signified by both literature and interview results. To achieve this, the interface communicates the system's current state, for example through the List-of-Waypoints component (Figure 5.13). Along with real-time progression on the map, the user sees what steps are ahead and what steps have been passed. Further, the tab bars visually represent mission phases (Planning, Flying, Landed) using colour-coding and text descriptions and where red highlights function as immediate warnings with accompanying text explanations for immediate feedback (Figure 5.14).



Figure 5.14: Tab bar with visuals and text showing the status of the flight

2. Match Between the System and Real World. In the design of the map layout, conventional components for map settings were employed. One matter that was discussed was the orientation of the map in relation to the drone. If only one drone is to be operated, having the drone centred and static on the map seems more intuitive as it mimics a first-person view. However, when multiple drones are viewed from an overhead perspective, the respective maps would adjust constantly to the static drones. Consequently, the decision was made to use static maps. Additionally, the navigator icon was designed to look like the actual SSRS drone to further adhere to this heuristic. Furthermore, when operating more than seven drones, the tab bar is divided into a dual tab bar. This structure was implemented in accordance with the approach of natural mapping, adhering to the heuristic.

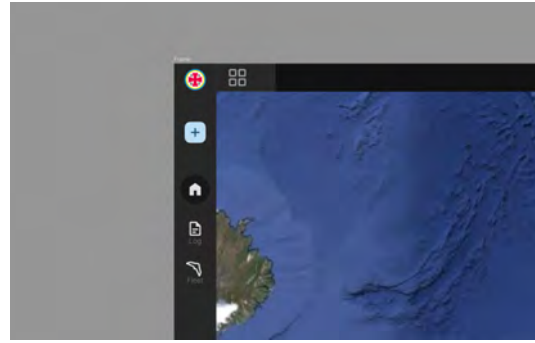
3. User Control and Freedom. Upon entering the coordinates, a new planning tab will automatically open in the tab bar, displaying the geographical location of the coordinates. This allows the user to determine the accuracy of the location. Should the location appear incorrect, the user has the option to close the tab in the same manner as a conventional browser tab functions. Adhering to the requirement “Interface adapted to experienced users” in Section 5.6, the interface will be used by experienced users, thus personalising and hotkeys can enhance the workflow. For undo and redo, conventional keyboard commands are employed. The potential for employing additional keyboard commands in order to facilitate even more shortcuts has been discussed as something for future work.

4. Consistency and Standards. The symbols for boats, aircraft and drones are collected from nautical charts and the ATC interface presented in Section 5.5, and are reused in the icons of the tab bar and map layer settings. The map functionality follows the navigational structure of Google Maps. The navigation rail follows material design guidelines structure and is an example where the interface uses external conventions (Figure 5.15).

5. Error Prevention. For this precise matter, the PUEA (Section 5.9) was conducted, where potential errors were identified and dealt with, resulting in further requirements. A matter deemed important was to prevent the user from accidentally launching the drone before the route is properly planned. Hence, resistance was incorporated into the launch buttons interaction, making it activate through press-and-hold for three seconds, avoiding accidental click. Another example of how error prevention has been applied is when the user plans a route over a restricted area or obstacles, the system signals it to the user by marking that section of the route red (Figure 5.16).



(a) Navigation rail according to Material Design 3. Image source: <https://m3.material.io/components/navigation-rail/guidelines>.



(b) Our navigation rail. Containing a primary action button and three main destinations. Along with the SSRS logo on the top.

Figure 5.15: Navigation rails compared



Figure 5.16: Altitude schematics component indicating a risk of collision ahead by marking the route red.

6. Recognition Rather than Recall. The user should not have to remember information from one part of the interface to another. In the case of controlling multiple drones simultaneously, operators need constant situational awareness to prioritize actions effectively. Switching between screens to find crucial data would hinder this process. To address this, we followed Ben Shneiderman’s principle “overview first, zoom and filter, details on demand.”, allowing us to prioritize the displayed information. By presenting an overview of all drones and their key statistics on a single screen, users can focus their attention efficiently without needing to recall details from other areas of the interface.

7. Flexibility and Efficiency of Use. Implemented in the design are keyboard shortcuts and a highly adaptable interface, which enable users to tailor their experience to their preferences and save it on their personal account. Additional features include window scaling and the ability to add/remove drone-specific widgets (Figure 5.17). However, there is considerable potential for further development in this area in future iterations of the interface.

8. Aesthetic and Minimalist Design. This one can be seen as complementary to the heuristic *Recognition Rather than Recall*. While the latter favours the presentation of information for direct recognition, the former focuses on the reduction of information. Notably, this aligns with the aforementioned concept Level of Detail (LOD) excessive information leads to information overload, while insufficient

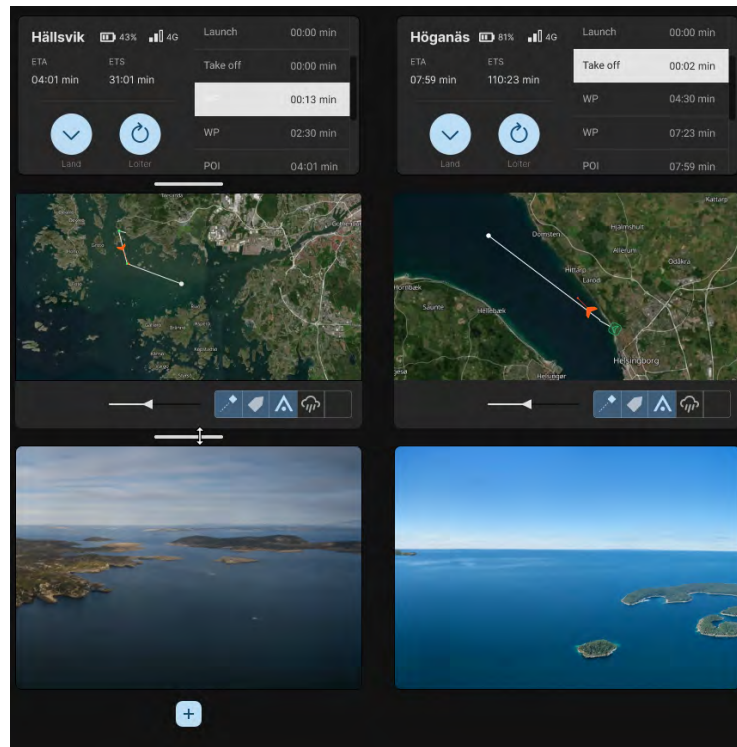


Figure 5.17: A section of the overview. The left column shows how customisation facilities appear while hovering over it. Lines indicate that the size of each window can be adjusted and the plus-icon allows the user to add other modules.

information can hinder decision-making. Aiming for a proper balance, the interface utilises an overview with only the most essential data, lined in a grid structure. This is aimed to guide the users eye vertically for comparing drone data and horizontally to focus on one specific drone. Further, the interface has a dark theme with a small variety of colours. It only uses colours where it symbolises something, like a red alarm in the tab bar, or a yellow boat on the map. Line weights are thin to separate components while not drawing attention from the user. Utilizing consistency in the spacing and grouping between components aimed to balance the aesthetics. This was achieved through design decisions such as colour palettes, line weights, and component spacing, resulting in an interface that is both aesthetically minimalist and functionally clear.

9. Help Users Recognize, Diagnose, and Recover from Errors. In a multi-control system, these messages are of critical importance for users' situational awareness and system effectiveness. Unclear error messages would hinder the system's ability to communicate with the user. Our approach employs the use of red colour for high visibility, attentive blinking, concise error descriptions, and actionable suggestions for resolution. The latter aligns with the automation management-by-consent approach and can be seen in Figure 5.18.

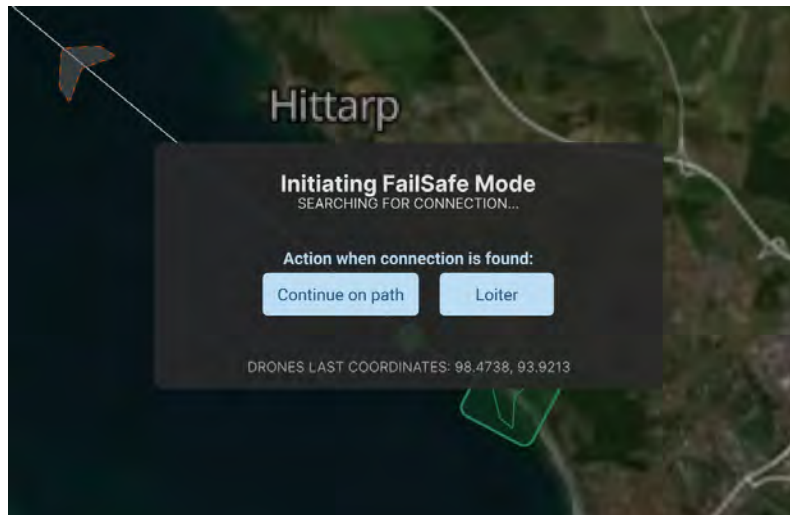


Figure 5.18: The dialogue that appears when the connection to a drone is lost is to help recognise and recover from an error.

10. Help and Documentation. Naturally, it is desirable that the interface should be intuitive enough to eliminate the need for separate user manuals. The intended user group for this interface are not novice users. Nevertheless, having access to proper information when needing help can be reassuring. Hence, there is a resource centre within the interface where documentation, user guides, regulations, and other important information will be kept. In response to feedback from the workshop, this section will also include quick access to NOTAM. This exemplifies the type of essential documents available within the interface, which minimises the need for external resources.

5.12 Qualitative Usability Testing

Following the development of a high-fidelity prototype, a usability evaluation (Section 4.4.1) was conducted. Ideally, usability testing involves participants who represent the targeted user group. However, in this project, a significant limitation was the absence of an established user group for SSRS drones, which could potentially impact the reliability of the findings. Our closest estimation of a suitable target group was people within SSRS with flight experience, and after meeting a segment of the intended target group, we considered ATCs to be a more appropriate user group of the DOS as they share qualities similar to those desired for multi-operational software. Hence, ATCs were asked, but no response was received. Instead, four participants were recruited from Chalmers University of Technology. They will be considered novice users and the evaluation was designed accordingly to provide valuable insights nevertheless.

5.12.1 Planning

To address the above limitation, and to gain comprehensive insights into user experience, the evaluation prioritised the collection of qualitative data over quantitative data. The latter typically rely on statistically significant sample sizes, and in our case, statistics would be misleading since they would derive from users without adequate experience. Thus working with qualitative data, the tests were designed to be in-depth, lasting approximately one hour each.

Participants were to walk through a series of tasks in the interface, each ending with subtasks of one – rapidly answer drone-specific questions displayed on the interface, and two – evaluate their personal experience so far. The walk-through followed the think-aloud protocol mentioned in Section 4.4.1. This combined approach aimed to gather rich qualitative data, even in the absence of a fully established user group. Furthermore, when creating the tests a target was to create an immersive experience for participants, simulating actually operating drones for SSRS. This aimed to align their experience and expressions more closely with that of the intended user group, thus addressing the aforementioned limitation by enhancing the reliability of the test results. In order to achieve this target, the test participants needed to be briefed with matters that would in reality be taught prior to becoming a drone operator. Thus, they were provided with a document containing the main goal of their work as well as field-specific details like acronyms, airspace regulations, and drone technicalities (Appendix G). This information was accessible to the participant throughout the test in order to evaluate the interface rather than the user's drone knowledge.

Aligned with the research question, the evaluation aimed to determine if or how the interface design supports multi-drone control as well as an estimated limit to the number of drones one user could effectively manage using the interface. To evaluate these matters, the tests were structured in an ascending manner – participants were to operate first one drone, then two, then three, and so on, until their perceived limit of what they could manage was reached. At each drone quantity, we compared their answering time to a set of questions to see if their answering time fluctuated with increasing the number of drones. Additionally, the semi-structured interviews (5.5) highlighted focus, stress and control as critical factors for multi-operations. By including these matters in differential scales, described in Section 4.4.2, the aim was to maintain a discussion with the participants on if and how their experience was affected by operating more drones. The tests were recorded, for the ability to re-visit them for analysis.

Prior to the evaluation day, a pilot test was conducted with one person. Feedback from the pilot study resulted in us modifying the evaluation structure, some specific questions as well as what to include in the reference guide. For example, explanation of the symbols representing aircraft, ships and static objects was not included in the reference guide during the pilot test and was added afterwards.

5.12.2 Protocol

At the beginning of the evaluation, participants were informed of the evaluation purpose and handed the reference guide. Moving forward, the participant was introduced to a scene:

“Today you are on duty as a drone pilot for the Swedish Sea Rescue Society. It is a beautiful day in August and several sea emergencies are expected.”

Starting the test, the participant was introduced to the home screen, and it was presented as the first thing the user sees when entering the site.

The first task was for the participant to explain what they saw and what features were presented. Subsequently, they received a call with coordinates to an emergency position, which they were instructed to send a drone to.

It was valuable for us to know if, after the presentation of the context, the user understood what was presented to them on the monitor and what their initial course of action in the event of an emergency call would be. When the call was received we observed if they were able to readily identify the button for initiating a new mission.

Their task was then to input the coordinates, identify an appropriate drone to launch, plan the route and send the drone on its way.

When selecting an appropriate drone, several factors must be considered in order to make an informed decision. The information they had learned were important to consider before launching a drone was presented on the screen (Figure 5.19). The objective of this task was to observe whether the participants were able to recognise the information and take it into account.

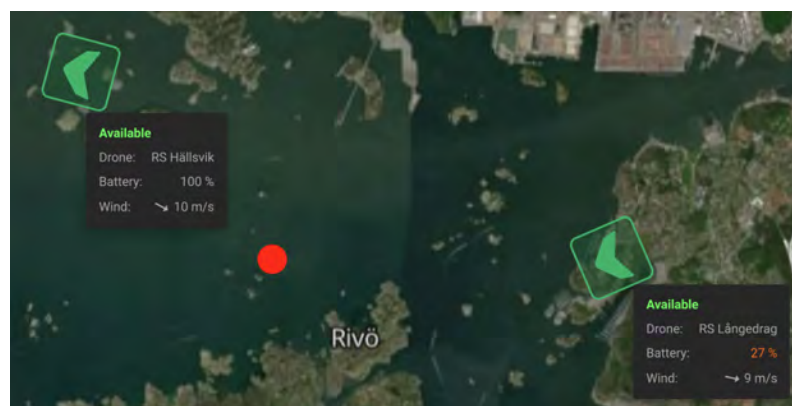


Figure 5.19: Before selecting a drone the user is presented with useful information about the drones in order for them to make an informed decision about which drone to select. The information is presented in the frame close to the drone’s position.

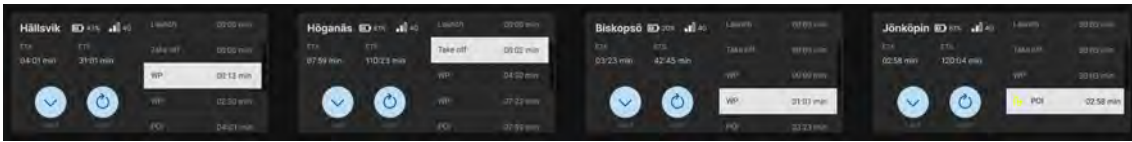


Figure 5.20: In the overview, the user finds information about the drones and their routes. When asking rapid questions to the evaluation participant, all answers could be found in this row of cards.

Thirdly, they were asked to navigate to the overview, where they could find all drones launched by them and where they were to answer specific questions about information that could be found in the overview, such as which of all drones was estimated to arrive first and the battery level of a specific drone.

As the overview was intended to provide cognitive relief to the user, by presenting all drones in a comprehensive summary, our aim with the design was to make it easy to access. By instructing the participant to enter the overview, we were able to indicate whether it was easy to find without specifying the exact location. In order to get a perception of when the participant had reached its limit in the amount of drones they were asked questions such as “When does the drone in Jönköping reach its destination?” and “Which drone has the highest battery level?”, information that could be found in the overview (Figure 5.20).

After the rapid questions, the participant was handed the three differential scales to fill in to reflect on their ability to accept one more mission. If so, the participants proceeded to perform the aforementioned tasks, launching drones and monitoring their drones in the overview until they themselves expressed a limit to the number of drones they could monitor. Ultimately, this structure allowed for the analysis of individual performance and subjective experience across different drone quantities. We could then identify the number of drones users could effectively manage within the interface and how the interface design influenced factors such as control, focus, and stress.

Other scenarios covered throughout the evaluation included an emergency situation located near a restricted area, in order to observe if the participants were able to identify the mark of a restricted area, and an emergency call requesting a search mission on scene, which would necessitate a higher endurance of the drone. The participants were therefore instructed to ensure sufficient time for the drone on scene before the its battery ran out. The evaluation was finalised with an interview, where we discussed why their differential scales were marked as they were, and how they changed over time.

5.12.3 Results

Upon analysis of each test, an affinity diagram was constructed for each individual test, and then merged into a single diagram. Themes and key factors were then collected from this merged diagram. Due to the extensive dialogue and interviews

conducted throughout the test, qualitative data could be derived from the test, despite the low amount of participants. The collective themes and insights are presented below.

Understanding the interface structure

Overall, the participants understood the interface and its functions well. They generally remembered the functions after using them once, pointing to the aim of *Recognition rather than recall* and could walk through all steps with the help of the field-specific document. Now focusing on interface improvement, insights on specific interface features will be addressed.

Two participants suggested the use of an additional screen. One proposed this to enable the operation of even more drones, while the other suggested it to display all video material on a separate screen. Further, two out of four participants felt it was unnecessary to know what the drone had done and preferred to focus solely on what was ahead. Continuing, they did not pay much attention to the altitude chart and assumed the drone was ready for launch. Adding to this, all participants found it difficult to press the launch button, which appeared to be caused by two reasons. Some mistakenly believed that the system prevented them from launching it, reflecting their trust in the system's automation. Others were affected by the knowledge that prototypes are not fully functional, which led them to *not* anticipate immediate feedback. Consequently, the error was not detected.

We conclude with a set of interface elements that may require revision:

- The drone fleet icon was misinterpreted.
- The overview icon was not immediately found and its functionality was misinterpreted.
- The selected tab was misinterpreted due to its dark colour.
- The low battery indicator was not noticed by all participants.

Feedback and system support

All participants preferred more system support than what was provided in the prototype. One participant noted, "The more drones there are, the more help I would say is needed. But at the same time, it must be the same whether you have one drone or seven drones" Another commented, "I find it a bit strange that [the system] considers crossing mathematically, then it is silly that it is even an option." All participants expressed a desire for clearer and more frequent warnings from the system. This was evident as some participants either did not notice or were slow to notice obstacles appearing on the map or in the camera view. One participant stated, "It should tell me which drone will go fastest." This might be due to the expectation that the system would provide warnings when necessary. As another

participant mentioned, “It’s not critically low battery because then it would have turned red, I assume” pointing to trust in the system. Furthermore, all participants expressed the need for clearer feedback when an action is executed. For example, “now this one is turning right” and “You have chosen for it to loiter at the found connection.” pointing to a desire for a transparent system. Ultimately, it is evident that the participants did not receive the level of feedback they desired from the interface which will be taken into account when determining the final results.

Monitoring multiple drones

The participants had the option to decline a drone mission when they felt like their limit had been reached. The individual limits for maximum number of drones operated were: five, six, seven and seven drones among the participants. When comparing their answering times across different quantities, it appeared that their mean times increased as they operated drones at their limit. For example, as seen in Figure 5.21, the participant had a slower answering time at seven drones and chose to decline the eighth drone mission. This pattern was observed for all participants (see Appendix H). It is also apparent that all participants had a slower answering time in the beginning which is assumed to be due to learnability matters within the test. From this, it is observed that none of the participants monitored more than seven drones, suggesting a potential limit to the number of drones an individual can operate simultaneously. However, due to certain limitations, such as the one-hour duration of the test (meaning participants may have grown bored of the testing, and not wanting to continue), as well as the small sample size, this cannot be concluded with certainty. Consequently, quantities over ten drones will not be explored further for the interface.



Figure 5.21: A chart of one participant’s mean answering time to the questions in relation to the number of drones they were operating. The first dot represents two drones simultaneously and the last dot represents seven drones simultaneously.

Two participants noted in the differential scales that their higher stress levels correlated with increased focus. One stated, “You gain more focus when more things are happening”, while another stated “You can’t be too calm in these situations; otherwise, you make mistakes”. Two participants expressed concerns that managing more than two drones would be challenging but still preceded to accept more drone missions. All participants indicated that the number of drones they could handle effectively would depend significantly on the situation. One participant commented, “Six calm drones are less troublesome than two with many urgent issues”. Two participants focused heavily on identifying which drones were most likely to need their

attention. These participants consistently checked all parameters and preferred to switch between full-screen views. One participant suggested the ability to make personal priority markings and write notes to keep track of all drones, particularly when managing a larger number. Two participants mentioned the importance of training, suggesting that the ability to manage multiple drones would be considerably easier with proper instruction. One participant noted that they could probably maintain a high level of focus for approximately two hours before needing a break.

Participants want to know where to direct their attention, to be able to feel in control. A way to accomplish this is to allow for the user to mark drones with certain markings, as suggested. This is similar to design patterns that can be found in for example video editing programs, web browsers and digital photo albums.

Human factors

A possible connection was observed between the participant's expressed discomfort in continuing to add more flights and their performance in the rapid questions while in the overview. Despite the participants' apparent readiness to continue adding drones, their comments indicated that they had reached their limit to how many drones they could manage already. For example, one participant expressed "It's becoming quite a lot to manage right now" while adding their fifth drone. This was subsequently confirmed by an investigation into their answering time and performance for the rapid questions. The fact that the participants continued to add more drones than they could manage could indicate that they were aware the program they were testing was a prototype. They understood that both the emergency scenarios and the flights were fictitious. If the scenarios were real, the participants would be expected to act differently. Primarily, they would feel more responsibility, which would result in a lower acceptance of the number of drone flights.

While not the focus of the evaluation, these personal attributes could affect the results. For instance, some participants demonstrated better memory retention throughout the test, whereas others forgot information quickly and desired the ability to revisit the walk-through. Furthermore, some participants were attentive to data changes within the interface, as one participant noted, "Now this drone has more battery [than it had before]". Some participants were more adept at noticing warning signs and obstacles, while others required prompts to identify warning signals. Additionally, one participant stood out due to their high pace, quick reaction time, and diligent checking of all drones simultaneously. In contrast, other participants needed reminders about the multiple drones in operation. This might suggest that the participant was highly engaged with the experience and focused intently on the test, or it could indicate that personal skill is crucial for managing multiple drones. This raises the question of the user group should be required to have certain skills and how the interface should be adapted to such expert users. Regardless, because people are different, the user should be able to personalise their workspace.

6

Result

This chapter presents the main results in response to the research question:

RQ: What should be considered when designing an interface for simultaneously operating multiple drones efficiently and safely in a sea rescue context?

First, in Section 6.1 we present nine considerations along with user experience factors that are of significance. These UX factors were developed as answer to sub-question 1: “What user experience factors are particularly important in the simultaneous operation of multiple drones in a sea rescue context?”.

Section 6.2 addresses how the considerations have informed the developed prototype through annotated frames. This prototype answers sub-question 2: “What solutions should be implemented in an interface to address the framework of design considerations?”

As mentioned in Section 1.3.3 this report is a contribution to design practitioners. Hence, the results presented in this chapter are concise in order for the designer to employ the considerations as guidelines when designing similar interfaces, where the prototype is displayed in the format of annotated frames. Further detail on decisions regarding the considerations and the design can be found in Section 7.3.1 and 7.3.2.

6.1 Considerations for the UI

The insights gained from all parts of the process have been distilled into a set of nine considerations, which are presented below. Additionally, each consideration is accompanied by specific user experience factors that are of significance. Having this in mind, the considerations for designing a safe and efficient GUI for multi-operation of drones in a sea rescue context are:

- C_1 **Mission adaptability.** Maritime rescue operations are dynamic and unpredictable. It is valuable for the system to take every possible scenario into account, however there is no way to know when everything has been factored

in. Hence, consider implementing user control in the form of high adjustability before and during flight, as circumstances and mission objectives may change over time, demanding the user to be aware of situational changes at all times.

UX factors: User control, Situational awareness

C_2 Reducing eligibility. Opt for an interface that is highly adapted towards the specific domain in order to streamline operations and cognitive processes. Sea rescue operations are dependent on fast response times, and the purpose of the drone is to be the first on the scene. Current interfaces are designed to accommodate a multitude of use cases, resulting in a multitude of options and selections available prior to launch. Consequently, the interface should be tailored for its purpose, hence does not have to take multiple use cases into account prior to the drone's launch. For sea rescue missions, different route options should not be included, the straightest line possible is to be preferred.

UX factors: Customisability and Personalisation, Cognitive load

C_3 Provide comfort to the user. The human factor is prominent within the system as the ultimate responsibility lies on the operator. Users' individuality should be accounted for by allowing for customisability in order to streamline personal workflow. Even though the user group should be trained experts, providing them with guides and regulation updates could relieve the cognitive workload. Consequently, it is recommended that guides and regulatory frameworks be integrated into the interface. Furthermore, even if emergencies rarely happen, the interviewees pointed to the comforting feeling of knowing that someone is at hand, remotely or physically. Human factors are inherently unpredictable, and temporary or situational disabilities may impair the operator's ability to perform at times. Hence, consider having a co-pilot to assist when the situation calls for it.

UX factors: Customisability and Personalisation, Cognitive load

C_4 Predict potential errors. Consider using the PUEA framework to identify potential use errors in the interface, and how to prevent them. The interface should be designed to inform the user about potential events to mitigate risks in the air or at sea. Additionally, consider what errors would be more effectively prevented through parts of the system other than the UI, such as with pre-training or through technical advancements in the drone.

*C*₅ **Layering Information.** When operating multiple drones on a single monitor, maintaining a good overview is essential for effective user focus distribution. Consider framing the overview on Schneidermans mantra “Overview first, zoom and filter, details on demand”. Furthermore, consider employing UX strategies such as Levels of Detail (LOD) and usability principles to avoid overload or lack of control. Utilize usability heuristics and conventional design patterns to achieve an intuitive and easily navigated interface.

UX factors: Cognitive load

*C*₆ **Careful Prioritisation.** When operating multiple drones, it is plausible that several drones will simultaneously demand the user’s attention. In such cases, the system should automatically prioritize drones based on parameters like time and risk. Additionally, it should allow users to manually prioritize drones based on external commands and to distinguish between them. Hence, establish a clear system for prioritizing drones that report issues with the vehicle or route to ensure timely intervention, while also enabling manual prioritization by the user.

UX factors: User control

*C*₇ **Strong Feedback.** In order to ensure that users are able to direct their attention to the correct component in a timely manner, it is necessary for the interface to be a strong communicator. This will enhance the user’s situational awareness, and thus enable them to prioritise their actions. To effectively attract the user’s focus, communicate warnings or errors using dynamic signals rather than static ones. Additionally, ensure that prioritized elements stand out by using a plain, minimalist interface design with bold warning signals for high contrast.

UX factor: Situational awareness, System transparency and Feedback

*C*₈ **Automation Level.** In the context of a cognitively demanding task, such as the monitoring of multiple drones simultaneously, a high degree of automation is associated with a reduced workload for the user. Nevertheless, excessive automation results in a reduction in manual work, as evidenced by the interviews, which identified this as a crucial factor in maintaining high levels of focus on the task. Regulations do not permit a fully automated system, as it is necessary for a human to be responsible for active decision-making. The management-by-consent approach, proposed by Ruff, Narayanan and Draper

[16], allows the system to provide suggestions to the user while allowing the user to make the final decision. It is recommended that the interface be designed in accordance with the management-by-consent approach, in order to achieve a balance between user control and cognitive load.

UX factors: User control, Cognitive load, System transparency and Feedback

*C*₉ **System Transparency.** For automated systems, the operator, who holds ultimate responsibility, needs to be aware of the drone’s activities. Therefore, it is important for the system to be transparent in its actions. Clear communication and feedback from the system facilitate situational awareness. The operator thus becomes aware of the drone’s capabilities and limitations, as well as those of the system. To achieve system transparency, it is recommended that the operator receives consistent feedback and clear information. Nonetheless, it is advisable to employ the principles of Level of Detail (LOD) to present clear information, thereby avoiding information overload.

UX factors: System transparency and Feedback

6.2 The Prototype

Throughout the project, a high fidelity prototype was developed and refined through iterations. The prototype is both a product of the design considerations outlined above and a means of evaluating the considerations at their early state. As a way of showcasing how the considerations were embodied, the prototype is presented below, in the format of an annotated portfolio (Section 4.4.3).

In general terms, this is a prototype of a web application designed for a computer screen with an aspect ratio of 16:9, additionally tailored to fit within the proportions of Google Chrome’s frame. The interface has been limited to a single screen, as it is believed that this simplifies user focus by consolidating all necessary information in one place. The interface has been designed to function within the SSRS organisation, where a remote pilot must deploy drones to accident sites in order to provide an initial visual overview to the rescue team through streamed material. The interface has been implemented in dark mode with the intention of facilitating prolonged user focus. The program is operated via mouse and keyboard. We have intentionally avoided excessive use of hotkeys, as this would divert the user’s attention from the screen to the keyboard. Nevertheless, we found it valuable to implement the commands “Undo”, “Redo”, “Loiter all drones”, and “Switching between drone tabs” as hotkeys, with the intention of streamlining the workflow.

6.2.1 Home screen

Figure 6.1 is a screenshot of the home screen. From here, and from all frames, the user has access to the navigation rail (1), the interface's top navigation, which contains three destinations: Home Screen, Operator's Log, and Drone Fleet Information. The Operator's Log updates the current operator on previous events, contains regulatory guides, and provides direct access to NOTAM. Adhering to C_3 , this approach relieves the user by gathering all relevant information in one accessible destination. This feature enables the pilot to prepare before receiving any emergency calls. The Drone Fleet Information contains logs about all drones within SSRS's fleet that inform about the drones' status, service updates as well as coordinates to their position. A circle in the navigation rail indicates the active destination. At the bottom is information about who is currently logged into the software (2), exemplifying a way to have a remote colleague on hand, and it is always visible throughout the interface as a comforting matter, adhering to C_3 . It also implements personal accounts to favor individual UI personalisation, further adhering to C_3 . At the top of the navigation rail is an add-button to start a mission (3), which is the primary action for this frame. When clicking on the add-button, an alert pops up in the middle of the screen for the user to type in the coordinates received from JRCC. This button is highlighted as we wanted the home screen to focus on the primary goal of the interface and minimise distractions, adhering to C_2 . Before any emergency call arrives, the user can interact with the map to explore current environmental parameters in order to familiarise themselves with current conditions. As seen in Figure 6.2 they may check the weather forecast, sea and air traffic, and static obstacles. Location predictors are attached to all moving objects, indicating their estimated positions at a set time, which can be adjusted using a slider in the map settings (4). The user can also switch the map type between satellite and flat view, zoom in and out, and pan around the map (5).

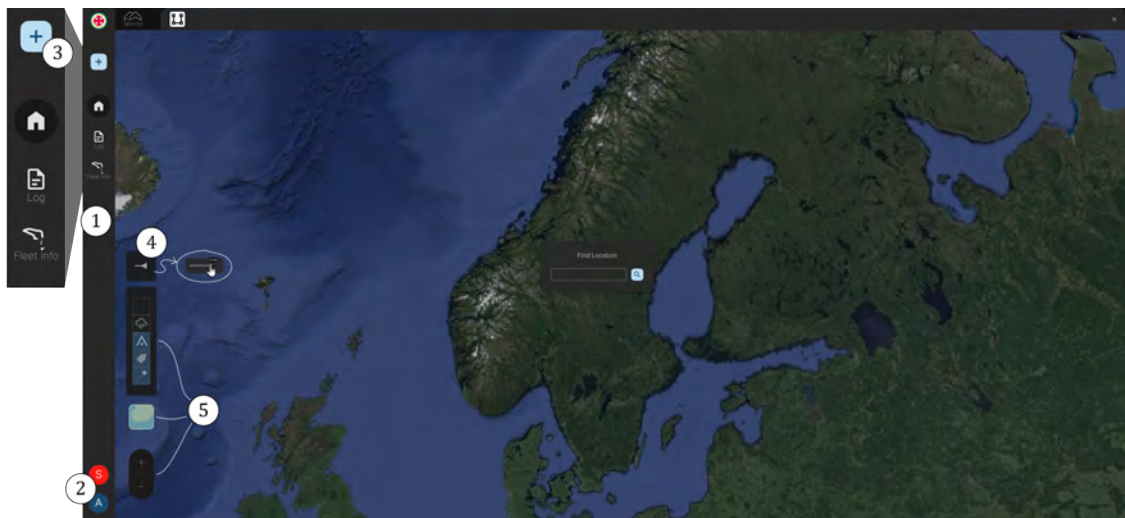


Figure 6.1: Home screen, the user is on its way to insert received coordinates to find the location on the map. Numbers in the image are referred to in the text.

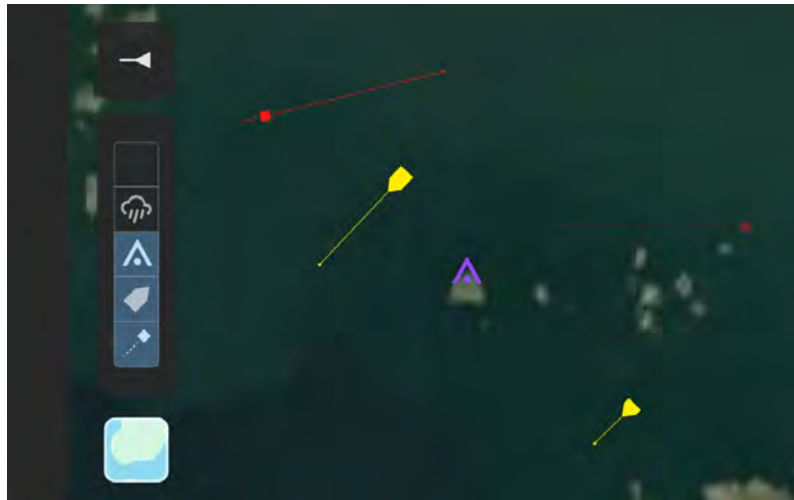


Figure 6.2: Image of map settings and corresponding elements in the map.



Figure 6.3: The targeted coordinates are showing on the map as a red dot with an animated red ring, representing “sending out signals”.

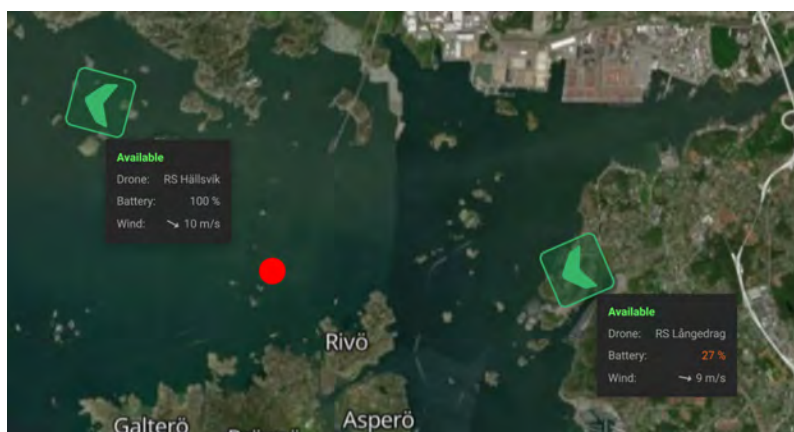


Figure 6.4: The inserted coordinates reveal a target on the map, showing as a red dot with an animated ring (as seen in Figure 6.3). The user has in this case two available drones to choose from to reach the target. They can select the most suitable drone for the mission.



Figure 6.5: The selected drone changes colour to orange.



(a) A route has been mapped out by the system. The system alerts the user about the island of larger size between the target and the launcher. The data about the island's ground level informs the user of a collision risk.

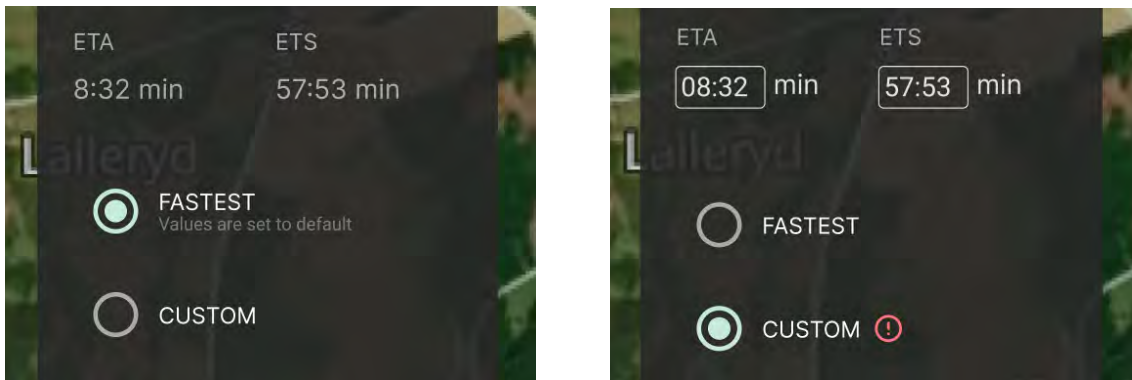


(b) The user updates the route.

Figure 6.6: Visualisation of how the system informs the user about inappropriate routes.

6.2.2 Planning Route

After entering the coordinates for the emergency location, the user is presented with nearby drones and can select the most appropriate one to launch (Figure 6.4). While the system could automatically select the optimal drone based on calculated variables, we chose to maintain user control by allowing the user to make this decision, adhering to C_8 . Once a drone is selected (Figure 6.5, a default route is plotted. Since the drone must launch against the wind direction, the take-off waypoint (WP) is placed accordingly, with a straight line route to the point of interest (POI). The user can modify the route by clicking and dragging on the line to add a new WP. In the scenario shown in Figure 6.6, the user needs to adjust the route due to the red indication line in the map and in the altitude chart. This exemplifies how we have balanced system support with user control, adhering to C_8 .



(a) When the duration at scene is not as important, the user should select the fastest speed to get out. “Fastest” is the default option.

(b) When the mission requires more time at scene than the fastest speed out can provide, the user selects “Custom” and can change the value in *Estimated Time at Scene* (ETS, before the battery is at 0 %). Since the drone will fly at a lower speed, the exclamation reminds the user to double check the *Estimated Time of Arrival* (ETA). If the ETA is too high, the user can alter that as well (in turn affecting the EST).

Figure 6.7: Module on the left side of the planning frame has been zoomed in to demonstrate how the user can adjust the speed in a comprehensible manner.

For most emergency calls reaching SSRS, the drone needs to arrive at the location as quickly as possible. The ETA (Estimated Time of Arrival) depends on wind conditions, distance to the target, and throttle amount. Achieving the fastest arrival time requires full throttle, which drains the battery more quickly, affecting the drone’s remaining flight time. However, in some rescue missions, the priority is to have more time at the scene, necessitating a higher battery level for endurance which can be achieved by flying at a lower speed to conserve battery life. In attempts to relieve the user from calculating these variables, the implementation can be seen in Figure 6.7. By default, the speed is set to “Fastest” since this is the most common scenario. If more time at the scene is needed, users can select “Custom”, making the ETA and ETS (Estimated Time at Scene) fields editable. Adjusting the ETS will recalculate the ETA based on a slower speed. This feature is tailored to the specific needs of the SSRS domain, adhering to C_2 . Additionally, these pre-configuration supports the user by excluding user adjustments on speed and throttle, providing only essential user needs, which adheres to C_2 and C_3 by reducing cognitive load and allowing the user to focus on the primary goal.

Connecting to the above, one of the purposes for having longer time at the scene could be for a search mission. This would require the drone to go on low speed in a search pattern until the target is found. This function is enabled by clicking on a WP and selecting search area (Figure 6.8). The alternative to move WP also shows up, enabling the user to promptly modify the route if needed, *adhering to* C_1 .

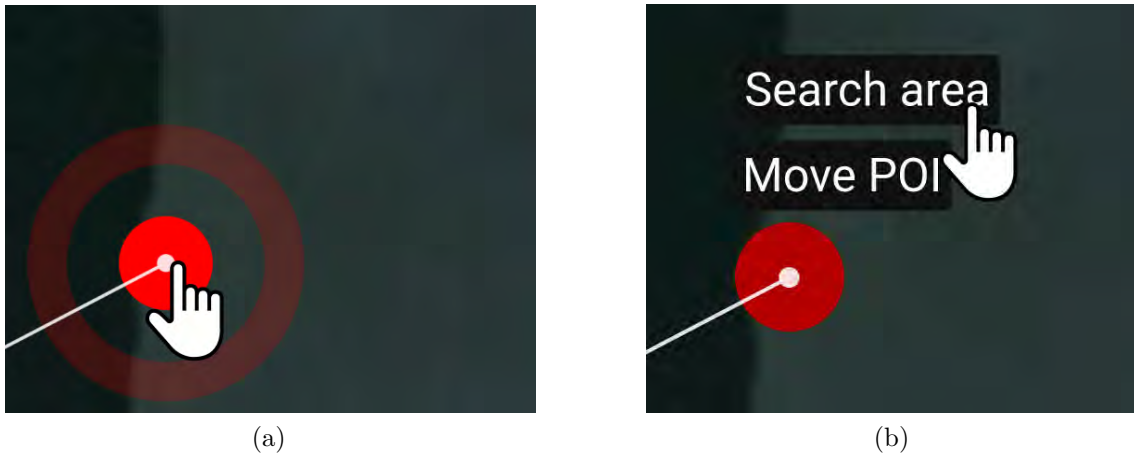


Figure 6.8: To create an area to do a search pattern, the user clicks the WP (a) and selects *Search area* (b).



Figure 6.9: Altitude schematics, alerting the user if there is a collision risk. The green line represents, in this figure, land and the red line is the section of the route which is in collision course.

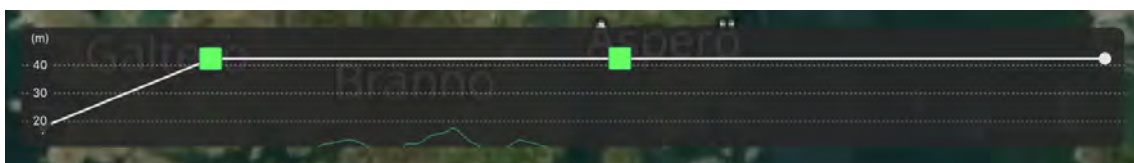


Figure 6.10: Altitude schematics visualising a clear course.



Figure 6.11: The launch is activated on a long-press on the launch button. To provide the user with feedback that the button needs to be pressed and held, an orange bar fills the button from left to right.

The altitude chart, shown in Figure 6.9 and 6.10, displays the topography corresponding to the route plotted on the map. This clearly indicates whether the path is clear. The drone flies at an altitude between 30 and 45 meters, and if the set route encounters an obstacle above this altitude, it is visualized with a red line in the altitude chart. In this scenario, the path is plotted over an island that is taller than the drone's set altitude, putting it on a collision course. Therefore, the user needs to either set a higher altitude for the drone or adjust the route to go around the island. Although this adjustment could be managed automatically in the background, displaying this information provides the user with control and promotes system trust, adhering to the system transparency mentioned in C_9 .

When the path is clear and the route has been modified to the user's preferences, it's time to launch the drone. The launch button is located in the bottom right corner and is activated by a long-press action (Figure 6.11). This placement and prolonged action help prevent accidental launches, adhering to error prevention as outlined in C_4 .

6.2.3 Operating One Drone

Once the drone is launched, the layout maintains a similar structure to the route planning interface but includes some different components.

Users can track the orange drone symbol on the map, ⑥ in Figure 6.12, to observe its flight to the target. The drone automatically parry against the wind to keep to the set route as closely as possible. The component known as the artificial horizon ⑦ provides direct feedback on how the drone is moving, utilising data from a gyroscopic sensor onboard the drone. While there was debate regarding the necessity of this component for users since the drone parry automatically, its inclusion addresses C_9 by providing transparency between the system and the user.

In the top-left component ⑧, the dynamic specifications of the drone are displayed as well as the name of the drone's corresponding SSRS station. Below are the direct actions the user can utilize on the drone, such as *Land*, which instructs the drone to shut off and land, *Loiter*, which directs the drone to circulate on the spot and



Figure 6.12: The frame during an active flight. The user sees the drone's position in the map and its altitude in the altitude schematics. The camera on the drone's image is visible in the bottom right corner. Numbers in the image are referred to in the text.

can be valuable as a means of “pausing” the drone, and *Right turn*, which prompts the drone to make a sharp 90-degree turn. The absence of a fast action for left turns is due to the application of the right-hand rule in airspace regulations. Figure 6.13 illustrates a scenario where the user has opted to have the drone loiter to allow an aircraft to pass and avoid collision. These direct action features illustrate the demand for user control to ensure safety in unforeseen circumstances, as addressed in C_1 .



Figure 6.13: When the drone needs to make way for other vehicles, one option is to immediately start loitering. The image shows how the interface informs the user that the drone is loitering.

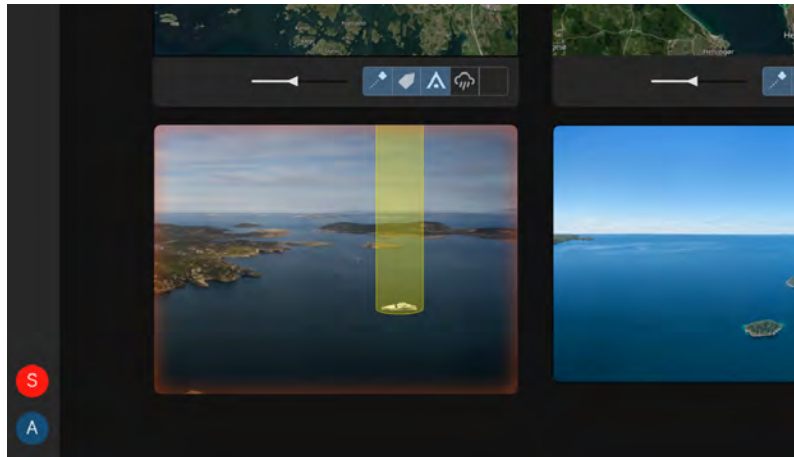


Figure 6.14: When a boat or other vehicle appears in the camera’s view, AI can identify the vehicle and inform the user that the air space above it should be avoided. It is visualised through a yellow cylinder in the frame.

During workshops and evaluations, it became apparent that the camera is less critical when flying to the target but can enhance situational awareness for the user. If something of interest is visible on the camera, the user can switch between the map and camera views using the symbol in the top left corner of the camera view. In instances where the user may not notice an obstacle in the camera view, it is anticipated that object detection can highlight these obstacles and provide warnings to the user, adhering to C_7 and exemplified in Figure 6.14.

Another way to detect obstacles is through the use of map layers. As outlined in the requirements table, the interface should nudge users to utilize map layers frequently to avoid missing moving obstacles. Therefore the obstacle map layers are selected by default, rather than the other way around. The map layers employ different symbols to represent obstacles at various heights, as illustrated in Figure 6.15. Drawing inspiration from ATC map structures, users can hover over specific objects for more detailed information. Adhering to C_5 we implemented transparency for airplanes at such high altitudes that they become irrelevant to the drone operator, as they are too far separated to pose any risk. The symbols design is an example of double coding where they are separated using both colour and shape, making them distinguishable for people with colour deficiency.

To the right of the screen is a list of all the WPs the drone will pass. Each row indicates its type (Launch, Take Off, WP, or POI) and the estimated time left until the drone reaches the WP. The time left to the POI is always the same as shown in the drone specs ETA. This list serves as a timeline for the drone and provides the user with an overview of the substeps remaining to the POI. The current WP that the drone is heading to is highlighted with a white fill. Clicking on a WP item expands it to reveal editable details on its individual altitude, as well as search pattern settings and a delete WP button. Layering of information adheres to C_5 .

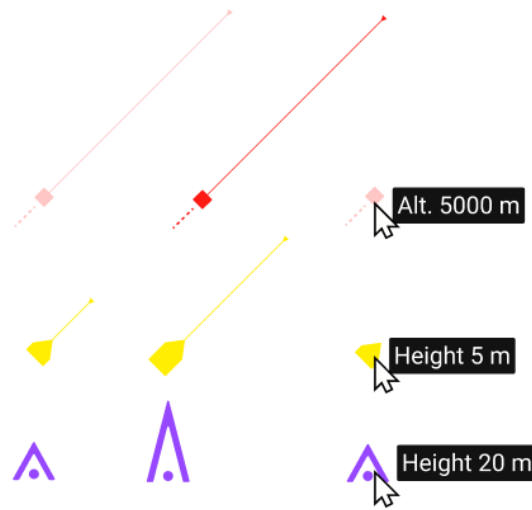


Figure 6.15: The symbols on the map change their appearance based on the obstacles' height or altitude. The more transparent red aircraft transponder indicates that the vehicle is at a high altitude. The user can therefore devote less attention to it. By hovering on the symbol, the user is provided with detailed information.

6.2.4 Monitoring Multiple Drones

To add another drone mission while operating an existing one, the user presses the add button, which generates a new tab in the tab bar. The setup process then follows the same steps as initially described. After launching the second drone, the user will now be operating two drones simultaneously. At this point, the user should navigate to the monitor tab to view an overview of all currently operated drones (Figure 6.16). The monitor tab acts as a landing page when the user is not actively managing a specific drone, providing a comprehensive outlook on all drones. From this point, the user's focus can be directed to the drone that requires immediate attention. The overview layout employs a grid structure, allowing for easy comparison of information across multiple drones. Each column in the grid represents an individual drone, with headings found in their respective tabs (Figure 6.17). Each tab contains the lowest level of information detail (Figure 6.18) displaying essential information – icon showing state, drone name, and estimated time to the next WP. The overview provided in the monitor tab is based on C_5 to achieve a balanced detail level on the information displayed. The tab bar remains visible across all screens, which we have leveraged to serve as warning communicators. With C_7 in mind, this should draw the user's attention from any frame in the interface (Figure 6.19).

If connection to one drone is lost, the user is notified and asked to make a decision regarding the next action when the connection is regained (Figure 6.20). This exemplifies the management-by-consent approach in C_8 . In the dialogue, the drone's last coordinates are shown. In accordance with C_6 , the user may wish to make personal notes for prioritisation or specific orders from JRCC. To facilitate this, the user can customise the interface by adding a notes module (Figure 6.21).

6. Result

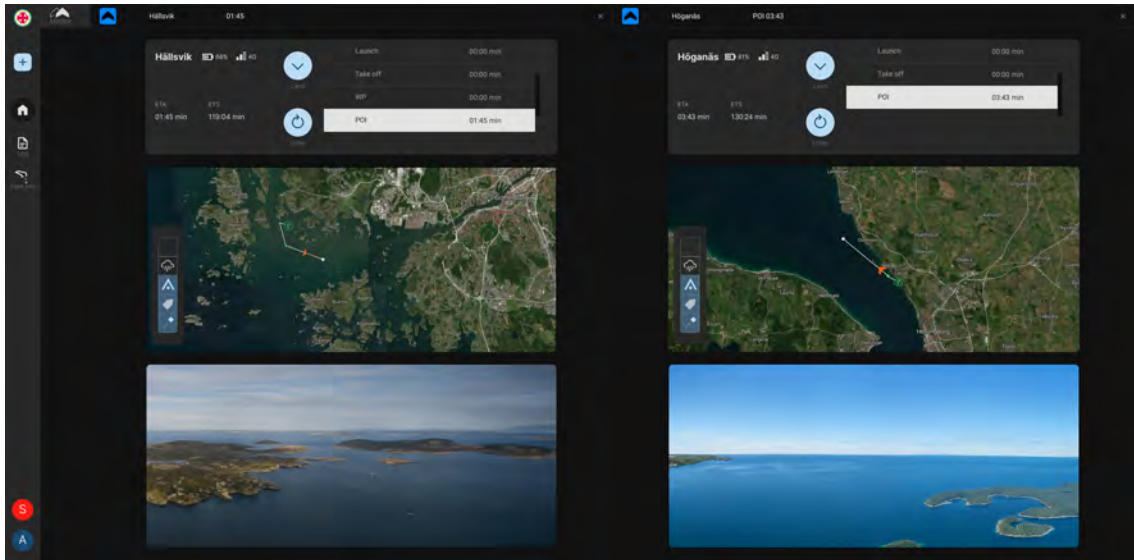


Figure 6.16: The overview when the user has two active flights. The information about each drone is collected in columns with corresponding tabs, which the user can click to view a larger map (Figure 6.12) and more detail about the drone and the environment.

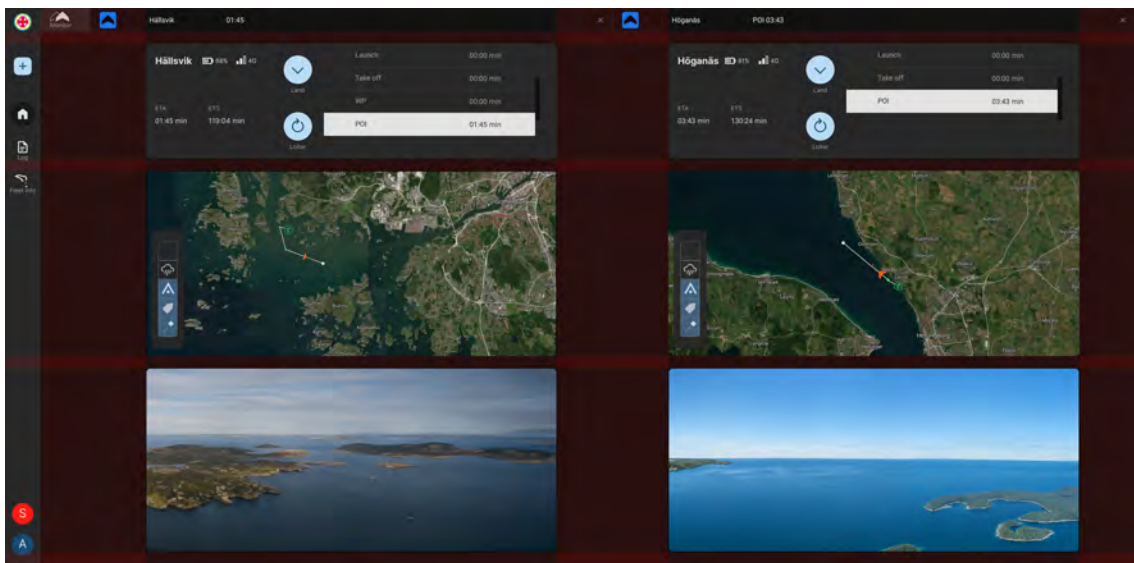


Figure 6.17: Each flight's information is collected in columns, and each type of data is presented in rows. The user can choose to focus on one flight, or one type of data (flight data, map, or camera) by checking columns or rows.



Figure 6.18: A tab leads the user to a more detailed view of a specific flight. The tab itself also contains information in a low level of detail (name of the drone, and estimated time to next waypoint).



Figure 6.19: The tabs are collected in a tab bar. In the figure, the left tab is highlighted in red, to indicate that the flight needs the user's attention. By highlighting a tab, the notice can be seen from anywhere in the interface.

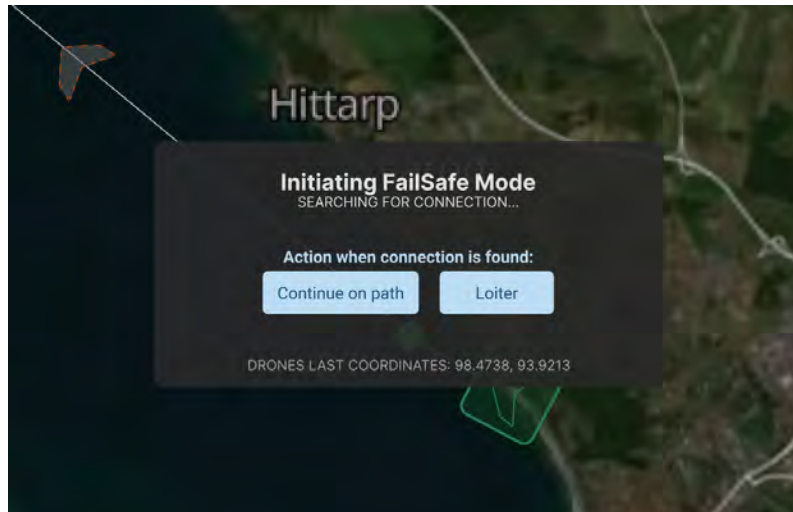


Figure 6.20: When connection is lost, the drone initiates a fail-safe mode, which the user will be informed about. The system suggests actions the drone can take when finding connection and the user can consent to one of them.

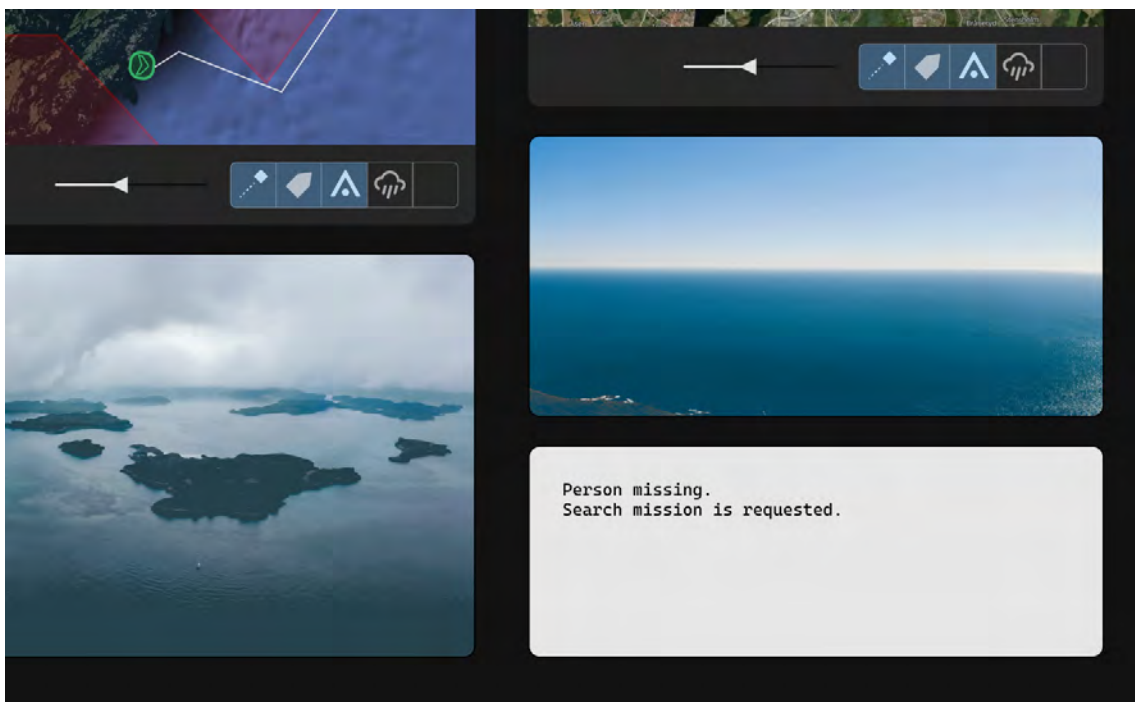


Figure 6.21: If the user wants to add a module, a text box is one of the modules that can be added to the overview.

6. Result

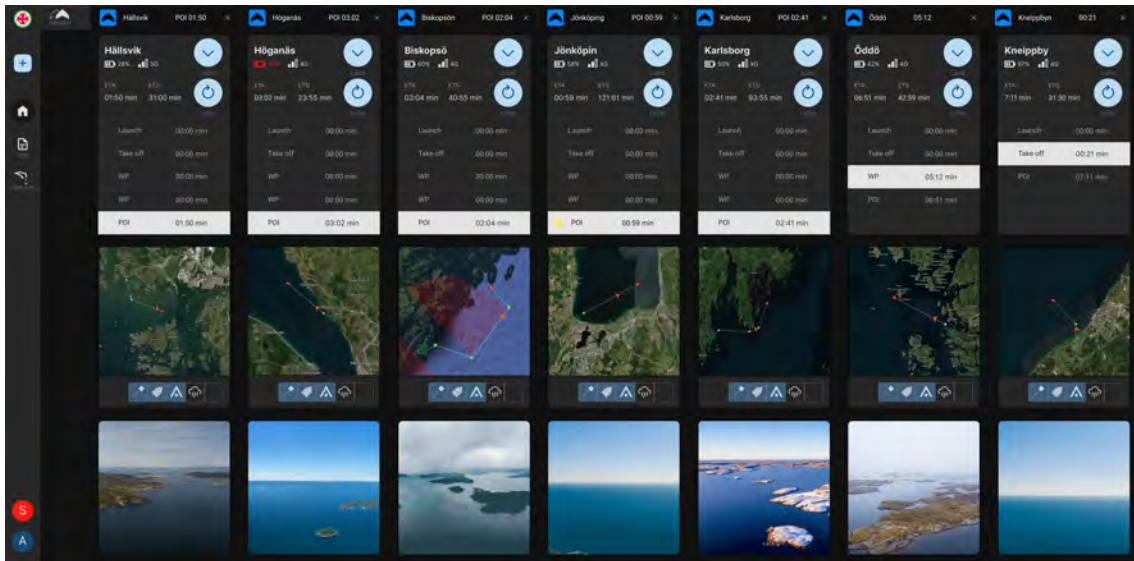


Figure 6.22: The overview when flying seven drones, the maximum amount of flights to monitor in a column view.

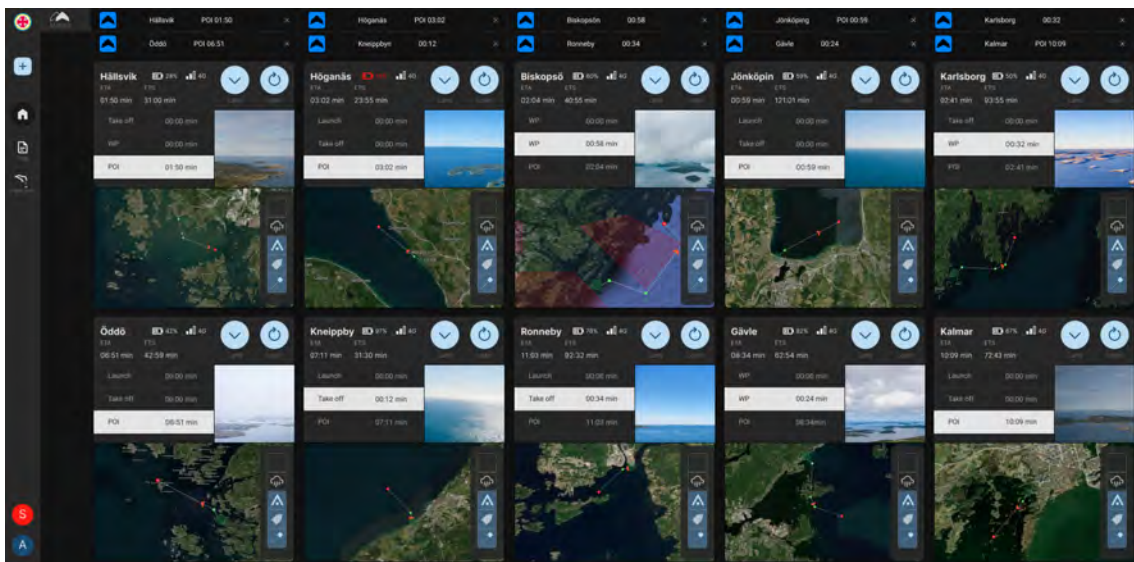


Figure 6.23: When flying eight or more drones, the overview changes structure from columns to a grid with two rows. The maximum amount of flights is ten.

Monitoring multiple drones on a single screen naturally involves the challenge of managing limited screen space. Figure 6.22 illustrates the layout accommodating up to seven drones within the interface. The range up to seven drones maintains a similar linear structure, where certain components adjust their layout in response to a reduced width of each column. Monitoring more than seven drones was considered the upper limit for the linear structure, necessitating a change to a stacked layout for 8-10 drones (Figure 6.23). Each drone card has a corresponding tab, and with the introduction of a stacked structure, a dual tab bar was implemented. This design decision adheres to the usability principle of natural mapping, aiming for intuitiveness even though the user's mental model of the interface may change significantly from seven to eight drones.

7

Discussion

The following chapter will comprise a discussion of the thesis, beginning with a discussion of the research question, the methodology employed, and the process of this project. It will proceed to discuss the results produced and finally it will conclude by discussing potential opportunities for future work in regards to the EOS-project and of the interface.

7.1 Execution and Process

Considering the results achieved out of this project, the research question was answered by providing a set of nine considerations (6.1) to inform GUI's for multi-operational drone applications in a sea rescue context. As answer to the first sub-question, the considerations highlight certain factors of user experience that are of significant importance to this specific domain, being: user control, customisability and personalisation, cognitive load, situational awareness and, system transparency and feedback. Answering the second sub-question, a prototype was made complemented with annotations to demonstrate how specific design patterns are implemented as suggested solutions to address the developed design considerations (6.2. Aligning with Gaver and Bowers's [39] suggestion, this is a way of contributing to the design domain. Further, our research contributes to the existing research gap (Section 2.4) within the field of drone development, namely the area of remote control of multiple drones at different locations. In particular, we provide considerations for a DOS interface, as well as an interface prototype for monitoring up to ten drones.

In line with RtD, mentioned in Section 3.2.2, our research contributions were derived through design iterations where design decisions informed the produced considerations and vice versa. To accomplish this, the project followed the Design Innovation Process. It is a dynamic model that allows for iterative design work, with no clear path to how a design process should be mapped. Figure 7.1 illustrates how the process has unfolded in the end, where one might distinguish between three main loops of iteration. The first one focused on knowing the topic which can be seen as it follows ①-②-④-③-②. This iteration comprised a comprehensive literature review, the creation of a storyboard, the undertaking of benchmarking exercises, and semi-structured interviews. These components were of critical importance, as

they established a foundational knowledge base for the remainder of the project. The literature was revisited, with new sources being added throughout the project as new areas of interest emerged. Storyboards enabled us to get an initial grasp of the target area and were revised as new insights to sea rescuing emerged. While the benchmarking process proved to be a valuable source of inspiration, the available software options were limited. A more accurate evaluation could have been achieved had the opportunity been available to test the software in actual drone flights. Furthermore, interviews were conducted with professionals from related fields. While these roles do not fully align with the remote drone operations, their contributions were valuable in gaining insights into potential user requirements and expectations. Their perspectives offered a novel viewpoint on the remote drone profession and the specific interface needs that such users might have.

The second iteration focused on framing and synthesising initial concepts following ④-⑤-②-③-④. It included Table of requirements, wireframes, wireflow, workshop, PUEA. The workshop included benchmarking evaluations, open discussions, and co-design ideation. However, it did not deliver the insights anticipated prior to the workshop. Participants appeared to have a negative attitude towards multi-drone operations. This may originate from their deep involvement in SSRS current sea rescue process, which may have prevented them from perceiving the potential benefits. Alternatively, the fact that all participants were former pilots may have caused them to associate multi-drone operations too closely with multi-aircraft operations. While the workshop did not contribute significantly to the concrete development of the interface, it did spark a valuable discussion on identifying the appropriate target group for the role and understanding the reasons behind the reluctance to trust multi-drone control. In retrospect, comparing the attitudes of pilots with those of air traffic controllers suggests that a mixed group might have provided a broader range of perspectives and more fruitful discussions. The insights gained from the workshop led to the modification of the storyboards, which, in conjunction with the error prevention strategies derived from the PUEA, resulted in the refinement of the wireframes. Overall, these methods proved effective, quickly establishing a foundation for the interface's development.

For the third iteration, we finalised a concept in high detail through evaluation which follows the steps ④-⑤-⑥-⑦-⑥-①. This iteration included Table of Requirements, Prototyping, Heuristic evaluation, Usability testing and Final refinements. A second draft of the Table of Requirements was established through the previous iteration. Guided by the wireframes, we began prototyping pixel-perfect components to be integrated into a reworked layout. Overall, these methods proved effective in reaching a final deliverable prototype. However, the usability testing revealed several limitations. Some of these could have been anticipated and addressed, while others highlight areas for future work. Firstly, the selection of participants could have been more closely aligned with the potential user group by, for instance, recruiting Air Traffic Controllers (ATCs). However, this might introduce other limitations due to their extensive experience and education. To accurately identify the ideal user group, a mix of different users and a higher quantity of tests would be preferable.

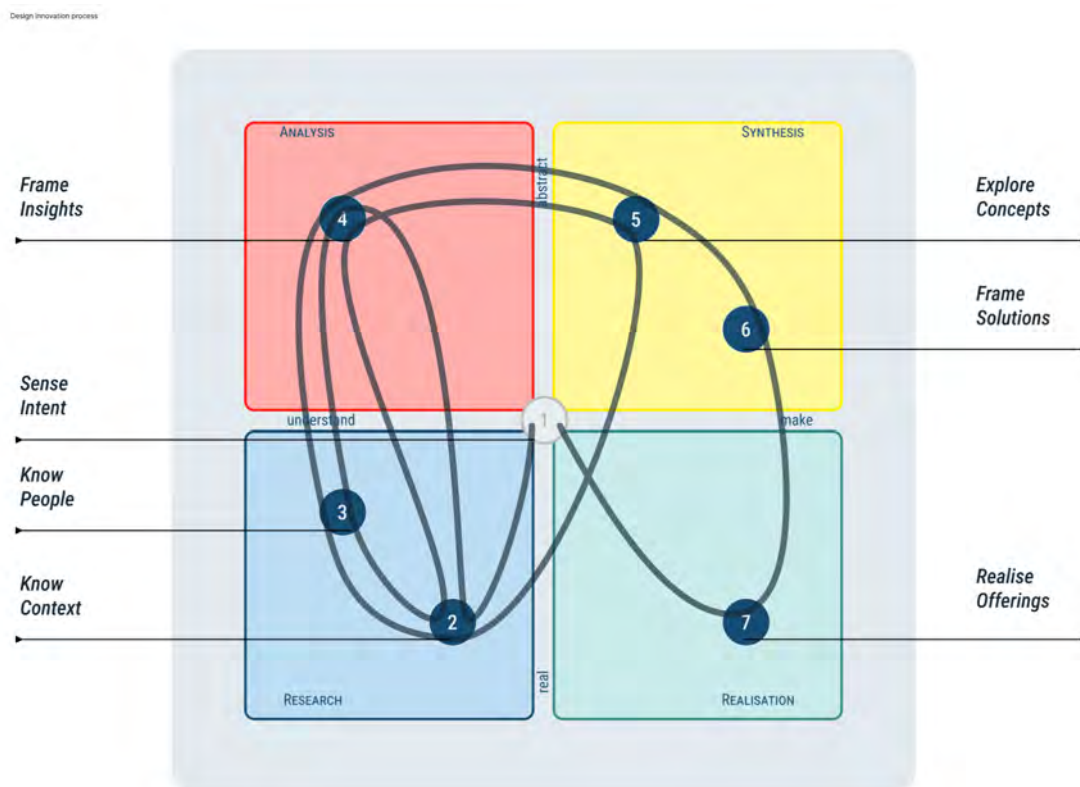


Figure 7.1: Illustration of our path in the Design Innovation Process.

Additionally, testing with more participants would allow for different scenarios and drone quantities to be evaluated, thereby ruling out irrelevant variables that may have affected the results. This could be accomplished by automating the evaluation, by setting up an evaluation station where curious potential participants could test the UI spontaneously. This would probably require a more compressed version of the evaluation in order to have the participants complete the test. This would collect more quantity data to add to the qualitative evaluation.

In summary, this kind of free design process model worked well for this project. Given that we had no clear design brief at the outset, meaning the balance of research contra synthesising phases was difficult to predict. Looking back, it is evident that the two left quadrants of the Design Innovation-matrix are heavier than the right ones. More work was spent on outlining the issue and framing needs for the interface. If further development of the interface were to continue, we purpose that the next iterations focus towards the right quadrants, emphasising evaluation and prototyping.

7.2 Ethical Considerations

This section will handle what ethical aspects considered in this thesis. Firstly, ethical considerations related to the process will be presented followed by considerations regarding the research contributions.

7.2.1 Informed Participant Consent

During the project we collected and stored personal data such as names, audio and video recordings, and images through interviews, and observations. We took the respondents' privacy in consideration and therefore, a participant consent form was written for this purpose. The form allowed the respondent or participant to check boxes asking if we are allowed to use names, profession, and photographs in the report. It also asked if the participant approve us to record and store audio and video material. The form described how the respondent can retract their approval of any of the points and that it can be done whenever they want to. In the form, we provided details about our project, its purpose, and about the data collected through interviews, observations, and usability tests was being stored in GDPR-certified OneDrive.

The project included usability tests on users. For these tests, the above mentioned consent form was filled out in order to ensure that personal information collecting was accepted and that the participants were aware of why and how their personal data was stored, and used. It was also clarified that during the tests, the participants skills were not being evaluated, only the prototype's ability to provide the users with desired content based on predefined scenarios. If an event would occur where a participant did no longer feel comfortable during a test, it could be immediately withdrawn. Imaginable scenarios were that this could be caused from stress when operating a drone or being overly immersed into the experience and thus claiming responsibility for clearing a life-or-death emergency.

7.2.2 Accessibility

Disabilities can be categorised in permanent, temporary and situational. To explain further what these categories mean, it can be exemplified with different types of hearing impairment. A permanent hearing disability is deafness, while an ear infection can cause temporary hearing impairment. A situational disability is caused by the person's surroundings, which can lead the person not hearing properly in a specific situation due to loud environments [43]. While user interfaces generally aim to be usable by the widest possible audience (a concept known as universal design), this project's targeted user group is relatively narrow, already with strict requirements placed on them: good vision, normal colour vision, normal hearing, and that they have passed cognitive selection tests [44], [45]. For this reason, and for not compromising too much on rescuing efficiency, no particular focus was on taking permanent disabilities into account. However, both temporary and situational disabilities can occur, and since the drone pilot has the ultimate responsibility of the

drone it was important to consider how to handle situations like this, keeping in mind guidelines from Microsoft's Inclusive 101 Guidebook [43] as well as Inclusive Design for Cognitive Exclusion Guidebook [46]. Designing for optimal visual clarity and minimal cognitive load was reached for during the process which potentially contributed to the over all user experience. For example, double coding was opted for when creating components in the interface to enhance visual clarity, which also is beneficial for people with color blindness.

7.2.3 Research Contributions

Our contribution in the field has the potential to increase the number of drones in society and have an impact on social and environmental factors. Current research indicates that drones will have a positive impact on the environment, as they have the potential to partially replace aircraft, which emit greenhouse gases, and take more efficient routes than ground vehicles. Additionally, our designed interface enables drones to assist sea rescue crews to choose a more sustainable vessel option when possible. However, there are risks associated with drone traffic, such as disturbing wildlife and natural areas with littering, sound pollution, and collision risks. To address the issue of littering, we have included the ability to fetch the drone's coordinates if lost, so that it can be found and collected. Another aspect that were considered was the social sustainability of drones within society. With the prevalence of drone delivery systems in urban areas, medical supplies and commercial goods can be delivered directly to customers' doors. However, there is debate as to whether such systems create new social constructs in society and whether the diminished need to leave home leads to isolation for some individuals. Finally, a concern with the increased presence of drones is their potential intrusion into citizens' privacy. This is a concern that exists today with the use of camera-equipped drones and their ability to capture footage that can induce a fear of being monitored by unknown authorities.

The interface makes images that are sometimes disturbing very accessible for a volunteer, which raises questions about the working conditions. In order to utilise the drones, the pilot (or someone designated to select images for the rescue crew) must have access to the footage from the scene. However, SSRS has expressed the drones as a 'nice-to-have' feature, which means that the drones are unlikely to be the difference between life and death in these missions, a factor which can reduce the stress of the drone operator. Regardless, the prospective user would be a member of the organisation, having already volunteered in lifesaving missions, which can be distressing at times in itself. Whether the drone operator should be a newly recruited volunteer or an already experienced crew member should be determined by SSRS.

Moreover, something that were considered is how our research contributes to the development of drones in other areas than for search and rescue. The thesis is public and may contribute to knowledge that can be used in other areas, such as the development of drones in military operations. Before the advent of commercial drones,

the term “unmanned aerial vehicle” (UAV) had a strong military connotation. To avoid military connotations, we used the term “drone” more frequently than “UAV” in this report.

7.3 Results

In response to the research question, we developed a set of considerations (Section 6.1) intended to inform the design of drone GUI’s and multi-control systems. In the same manner, the considerations informed the development of our own design (Section 6.2). This section will discuss certain design decisions and consideration aspects.

7.3.1 The Considerations and UX Factors

The considerations are derived from various aspects of this thesis. Most considerations emerged from multiple stages of the process and were refined or strengthened as we employed different methods. For instance, C_5 **Layering Information** was initially drafted based on literature insights concerning related work and the Level of Detail. Its importance was further emphasized during interviews and prototype evaluations. In contrast, C_4 **Predict Potential Errors** was primarily derived from the PUEA method, which might suggest less reliability. However, in the context of maritime rescue operations and airspace safety, this issue was deemed to be of significant importance.

A point for discussion is the synergy among the considerations. The considerations C_3 **Provide Comfort to the User**, C_5 **Layering Information**, and C_7 **Strong Feedback** undoubtedly complement each other well, all aiming to reduce the user’s cognitive load and create a visually clear interface. However, considerations C_1 **Mission Adaptability**, C_2 **Reduce Eligibility**, and C_8 **Automation Level** might initially appear contradictory. C_1 advocates for providing user control due to the unpredictability of SAR missions, while C_2 suggests limiting user choices to reduce cognitive overload and streamline workflow. Balancing these seemingly conflicting considerations is crucial. Options should be available, but the interface must be adapted to the specific use case to enhance efficiency. The balance between these considerations ultimately hinges on effective information layering and prioritisation, as covered in the other considerations.

Additionally, some considerations, such as C_7 **Strong Feedback** or C_9 **System Transparency**, might seem self-evident to designers. However, given the level of autonomy and feedback from evaluations, they were deemed vital for effectively monitoring multiple drones simultaneously, directly relating to the research question, we decided to include them.

To recall from Section 6.1, the considerations highlight certain areas of user experience, where a UX factor can be found in several considerations and a consideration can address several UX factors. *User control* is highlighted in C_1 , C_6 and

C₈. Customisability and Personalisation is highlighted in *C₂* and *C₃*. *Cognitive load* is highlighted in *C₂*, *C₃*, *C₅* and *C₈*. *Situational Awareness* is highlighted in *C₁* and *C₇*. Finally, *System Transparency and Feedback* is highlighted in *C₇*, *C₈* and *C₉*. Evidently, certain factors are addressed in more considerations than others, indicating that they may be of greater importance when designing such UI. If so, Cognitive load may be seen as the most prominent factor, as it is mentioned in most considerations. On the opposite, Situational awareness and, Customisability and Personalisation are only mentioned in two considerations. The study by Niroui, Liu, Bichay *et al.* [15] found reducing cognitive load and situational awareness to be the most crucial factors of a multi-robot control. While Situational awareness is still to be considered important, it may be of lower significance in our study than in Niroui, Liu, Bichay *et al.*'s due to the application differences. On a final note, these user experience factors are aims of the interface. But as mentioned in Section 3.1.1 “user experience cannot be designed, but it can be designed for” [19] meaning that the ultimate achievement of these factors are uncertain until further evaluations can be conducted.

7.3.2 The Design

The interface prototype was a way of embodying the aforementioned considerations. It is important to note that the design presented is just one possible application of these considerations. Without evaluation in a real-world setting, it is plausible that other interfaces might offer better implementations. One significant finding was the impact automation level has on the design. Research suggested that the management-by-consent level is appropriate for operating multiple units simultaneously, as it balances user control with cognitive processes. However, for this specific context, it remains uncertain whether this level is the most suitable. Moreover, specific design patterns, such as map orientation and the component artificial horizon, were subjects of discussion. In the current design, the map is static, and the drone moves along its route. An alternative approach could be to keep the drone static and adjust the map as the drone turns. As discussed in the heuristic evaluation section (4.3.7), the chosen approach avoids the confusion of moving maps when the user is in monitor view, however, this decision would benefit from further investigation. Similarly, the inclusion of the artificial horizon was discussed with the client. It is a component used by pilots, thus its necessity to a stationary drone operator is debatable. The benchmarking indicated its importance, and it was considered valuable for enhancing system transparency and user control by allowing users to visually see the drone's actions. However, it is possible that an alternative component for addressing these concerns would be more suitable for a drone interface. Finally, the interface was initially designed for use on a single computer monitor. This decision aimed to maintain a static focus for users and increase the practicality of remote drone operation within SSRS's organisation, as it does not place undue demands on specific equipment. However, two out of four participants expressed a preference for an additional monitor, either for increasing the capacity to manage more drones or to display all video streams on a separate monitor. Hence, this aspect may need reconsideration.

7.4 Future work

At the outset of the project, the target group was defined as individuals with experience in sea rescue operations, with a background in flying or operating airplanes or drones. This group included ATCs, pilots, and drone pilots within SSRS. During the workshop with solely pilots, it was clear that flying more than one aircraft simultaneously was perceived as an excessive workload. The pilots are accustomed to flying only one machine, which could be a contributing factor to this attitude. In the interviews however, the attitude differed from that of the participants in the workshop. The interviewees were all more familiar with the concept of autonomous operating and as the project progressed, it became evident that the group should be further narrowed down, with ATCs serving as the primary focus. However, the profession is not particularly common, with approximately 700 air traffic controllers in Sweden. This results in them being difficult to reach. Despite this, we were able to contact one, who proved to be a valuable contributor to the project. This experience led us to conclude that for further research, a more suitable target group to consider would be ATCs specifically.

In Qualitative User Testing (section 5.12) it is mentioned that the participants would appreciate the system offering more automated suggestions. This could result in a reduction in their level of focus, allowing the user to be available for a longer period of time as the work is not too strenuous. However, an operator who is too unfocused and inactive is slower to react to unexpected events. When searching for a good balance, the question arose as to how important it really is for the user to maintain high focus while monitoring the drones, if the system is automated. In the event of an incident, the drone should be capable of autonomously responding and resolving the situation (without the need of the user reacting fast, or reacting at all). During the workshop participants expressed a worry about the transition from manual steering to automated systems, which may imply that they did not trust the automated systems fully. According to Wang [17], the more automated systems become, the more it needs to be transparent and inform the user about decisions, in order for the user to rely on the system. The current regulations are clear however, the drone operator has the ultimate responsibility, as mentioned in Section 2.1.1, and therefore it is not compatible with the drone making its own decisions. The final interface for this project was designed to align with Ruff, Narayanan and Draper's [16] suggested autonomy level management-by-consent, which represents a higher level of autonomy where the user maintains control. Should regulations and attitudes towards technology evolve in the future, a further increase in the level of autonomy could possibly be applied in the future. This could enable the operator to focus more on other tasks related to the drone mission, such as sending representative images of the emergency situation to the rescue crew.

All instances in the rescue mission has connection to the radio communication tool *Rakel* and a drone operator is likely to be participating in that radio call as well. According to the workshop participants, these radio channels have the potential to become highly conversational. When the drone operator monitors several drones

simultaneously, this aspect of the project becomes more complex. It is necessary to determine whether the drone pilot is able to listen to multiple radios simultaneously. Furthermore, it is essential to ascertain how the pilot could be enabled to distinguish between the voices of different missions. It may be beneficial to divide the audio in order to facilitate the pilot's ability to distinguish between the various missions. However, this raises the question of how the pilot is to know which radio channel to direct its focus to. Another aspect that requires consideration is the potential application of a speech-to-text approach. This could enable the user to assimilate instructions and other messages from the crew on the boat. It is necessary to determine whether it is essential for the pilot to be able to communicate back or whether a one-way connection is sufficient for the purpose. This issue is one of many parameters that needs to be solved before realising multi-operation of drones. While the solution is beyond the scope of this project, and more aligned with the HDI category *Enhancing human-drone communication* mentioned in Section 3.3, it could ultimately impact the interface and is a valid area for future work.

Furthermore, developing the prototype into fully programmed software with complete interaction possibilities would serve as a foundation for user tests which likely would make participants more immersed in the simulation, believing they were operating real drones, hence produce reliable results for refinement of the interface.

8

Conclusion

The aim of this Masters thesis project was to answer the research question: “What should be considered when designing an interface for simultaneously operating multiple drones efficiently and safely in a sea rescue context?”, in contribution to practitioners in design or HDI with similar intents. The project was carried out in collaboration with the Swedish Sea Rescue Society as a part of their project Eyes-On-Scene, which has the ultimate objective of deploying a camera equipped drone to an emergency site in order to send footage to the rescue crew, which in turn is hoped to streamline rescue processes. Although current regulations do not permit such activities, this project brief was developed to explore the feasibility of remotely operating multiple drones by a single operator. To approach this, the project followed the Design Innovation Process where phases of research, analysis, synthesis, and realisation guided the iterations of interaction design methodology, including HDI research, interviews with pilots and ATCs, wireframing, prototyping and culminating in usability testing on the high-fidelity prototype. Findings concluded a set of nine considerations (Section 6.1) in response to the RQ, being; Mission Adaptability, Reducing Eligibility, Provide Comfort to the User, Predict potential Errors, Layering Information, Careful Prioritisation, Strong Feedback, Automation Level, System Transparency. The considerations highlight certain areas of UX (user control, customisability and personalisation, cognitive load, situational awareness, and, system transparency and feedback) that impact the user’s experience. Along with the considerations, a high-fidelity prototype was developed during the design process and used to formulate and evaluate considerations, which in turn guided the design of the prototype along with annotations as an additional research contribution (Section 6.2). As discussed in Section 7.4, further work is needed to validate the prototype’s ability to enable efficient and safe operation of multiple drones. This would require implementing the design in code to conduct proper test flights with the interface. Additionally, the project faced the limitation of not having an established user group, which may have influenced the evaluation of the prototype and the formulation of considerations. Nevertheless, we believe the insights gathered during this project are sufficient to answer the research question through the formulated considerations. Together with the annotated prototype, these deliverables should provide valuable knowledge to inform design decisions for a multi-control drone GUI in a sea rescue context.

Bibliography

- [1] Sjöfartsverket, “Årsstatistik 2023 för sjö-och flygräddningstjänst samt helikopterinsatser,” Tech. Rep., 2024.
- [2] M. Miron, D. Whetham, M. Auzanneau and A. Hill, “Public Drone Perception,” *Technology in Society*, vol. 73, p. 102 246, May 2023, ISSN: 0160-791X. DOI: 10.1016/J.TECHSOC.2023.102246.
- [3] P. Kardasz, J. Duskocz, M. Hejduk, P. Wiejkut and H. Zarzycki, “Drones and Possibilities of Their Using,” *Journal of Civil & Environmental Engineering*, vol. 6, no. 3, 2016. DOI: 10.4172/2165-784X.1000233. [Online]. Available: <http://dx.doi.org/10.4172/2165-784X.1000233>.
- [4] M. McNabb, *The Evolving Benefits of Drones in Search and Rescue: From the Floor of ElevateUAV*, Jul. 2023. [Online]. Available: <https://dronelife.com/2023/07/26/the-evolving-benefits-of-drones-in-search-and-rescue/>.
- [5] S. Leon, C. Chen and A. Ratcliffe, “Consumers perceptions of last mile drone delivery,” *International Journal of Logistics Research and Applications*, vol. 26, no. 3, pp. 345–364, Mar. 2023, ISSN: 1469848X. DOI: 10.1080/13675567.2021.1957803. [Online]. Available: <https://www.tandfonline.com/doi/abs/10.1080/13675567.2021.1957803>.
- [6] M. A. Goodrich, B. S. Morse, D. Gerhardt *et al.*, “Supporting wilderness search and rescue using a camera-equipped mini UAV,” *Journal of Field Robotics*, vol. 25, no. 1-2, pp. 89–110, Jan. 2008, ISSN: 15564959. DOI: 10.1002/ROB.20226.
- [7] N. D. Rasmussen, B. S. Morse and C. N. Taylor, “A fixed-wing, mini-UAV system for aerial search operations,” *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference 2007*, vol. 5, pp. 4743–4750, 2007. DOI: 10.2514/6.2007-6819. [Online]. Available: <https://arc.aiaa.org/doi/10.2514/6.2007-6819>.
- [8] *Drones to support sea rescue operations*. [Online]. Available: <https://altigator.com/en/drones-to-support-sea-rescue-operations/>.
- [9] M. Nafiz, H. Khan and C. Neustaedter, “Exploring Drones to Assist Firefighters During Emergencies,” May 2019. [Online]. Available: <https://hal.science/hal-02128386%20https://hal.science/hal-02128386/document>.
- [10] F. Falkman, *Personal contact*, Gothenburg, 2024.
- [11] Sjöfartsverket, *Årsstatistik för sjö- och flygräddningstjänst 2022*, 2023.

- [12] “Vårt uppdrag - Sjöräddningssällskapet,” Tech. Rep., pp. 28–28. [Online]. Available: <https://www.sjoraddning.se/om-oss/vart-uppdrag>.
- [13] Sjöfartsverket, “Sommarstatistik för sjö- och flygräddningstjänst 2023,” Tech. Rep., 2024.
- [14] M. Langerwisch, T. Wittmann, S. Thamke, T. Remmersmann, A. Tiderko and B. Wagner, “Heterogeneous teams of unmanned ground and aerial robots for reconnaissance and surveillance - A field experiment,” in *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, IEEE, Oct. 2013, pp. 1–6, ISBN: 978-1-4799-0880-6. DOI: 10.1109/SSRR.2013.6719320.
- [15] F. Niroui, Y. Liu, R. Bichay *et al.*, “A graphical user interface for multi-robot control in urban search and rescue applications,” *IRIS 2016 - 2016 IEEE 4th International Symposium on Robotics and Intelligent Sensors: Empowering Robots with Smart Sensors*, pp. 217–222, Oct. 2017. DOI: 10.1109/IRIS.2016.8066094. [Online]. Available: <https://ieeexplore.ieee.org/document/8066094>.
- [16] H. A. Ruff, S. Narayanan and M. H. Draper, “Human interaction with levels of automation and decision-aid fidelity in the supervisory control of multiple simulated unmanned air vehicles,” *Presence: Teleoperators and Virtual Environments*, vol. 11, no. 4, pp. 335–351, Aug. 2002, ISSN: 10547460. DOI: 10.1162/105474602760204264.
- [17] T. Wang, *Transparency, trust, and level of detail in user interface design for human autonomy teaming*, Nov. 2023. [Online]. Available: <http://hdl.handle.net/10919/116632>.
- [18] *What Are User Flows? | IxDF*. [Online]. Available: <https://www.interaction-design.org/literature/topics/user-flows>.
- [19] H. Sharp, J. Preece and Y. Rogers, “Interaction Design: beyond the human-computer interaction,” vol. 1, no. 4, p. 657, 2018.
- [20] M. Hassenzahl, “The Thing and I (Summer of 17 Remix),” pp. 17–31, 2018, ISSN: 2524-4477. DOI: 10.1007/978-3-319-68213-6_{_}2. [Online]. Available: https://link-springer-com.proxy.lib.chalmers.se/chapter/10.1007/978-3-319-68213-6_2.
- [21] V. Kumar, *101 Design Methods*. Nov. 2018. [Online]. Available: https://www.researchgate.net/publication/328861557_101_Design_Methods.
- [22] C. Frayling, “Research in Art and Design,” *Royal College of Art Research Papers*, vol. 1, no. 1, pp. 1–5, 1993.
- [23] J. Zimmerman and J. Forlizzi, “Research Through Design in HCI,” in *Ways of Knowing in HCI*, New York, NY: Springer New York, 2014, pp. 167–189. DOI: 10.1007/978-1-4939-0378-8_{_}8.
- [24] B. Martin and B. Hanington, *Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions*, 58480th ed. Rockport Publishers, Feb. 2012.
- [25] W. Gaver, “What should we expect from research through design?” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA: ACM, May 2012, pp. 937–946, ISBN: 9781450310154. DOI: 10.1145/2207676.2208538.

-
- [26] D. Tezza, M. A. .-. I. Access and u. 2019, “The state-of-the-art of human-drone interaction: A survey,” *ieeexplore.ieee.org*, vol. 7, 2019. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/8903295/>.
- [27] K. Cho, M. Cho and J. Jeon, “Fly a Drone Safely: Evaluation of an Embodied Egocentric Drone Controller Interface,” *Interacting with Computers*, vol. 29, no. 3, pp. 345–354, 2017. DOI: 10.1093/iwc/iww027. [Online]. Available: <https://academic.oup.com/iwc/article/29/3/345/2607841>.
- [28] *What are User Interface (UI) Design Patterns? updated 2024 | IxDF*. [Online]. Available: <https://www.interaction-design.org/literature/topics/ui-design-patterns>.
- [29] A. Joyce, *7 Steps to Benchmark Your Products UX*, 2020. [Online]. Available: <https://www.nngroup.com/articles/product-ux-benchmarks/>.
- [30] M. Tomitsch, C. Wrigley, M. Borthwick *et al.*, *Design. Think. Make. Break. Repeat. A handbook of methods*, Revised. Amsterdam: BIS Publishers, 2020, ISBN: 978-90-6369-479-1. [Online]. Available: <http://www.bispublishers.com/design-think-make-break-repeat.html>.
- [31] Miro, *What Is An Affinity Diagram And How Do You Use It? | MiroBlog*, 2023. [Online]. Available: <https://miro.com/blog/create-affinity-diagrams/>.
- [32] L. Leighfield, *How to Storyboard: Step-by-Step Storyboarding Guide (2024) | Boords*, Dec. 2023. [Online]. Available: <https://boords.com/how-to-storyboard>.
- [33] *Wireframe vs. Mockup vs. Prototype: What Is the Difference? | Aha! software*. [Online]. Available: <https://www.aha.io/roadmapping/guide/product-management/wireframe-mockup-prototype>.
- [34] Louise Bruton, *A complete guide to prototyping - UX Design Institute*, Jul. 2022. [Online]. Available: <https://www.uxdesigninstitute.com/blog/prototyping-guide/>.
- [35] J. Nielsen and R. Molich, “Heuristic evaluation of user interfaces,” in *Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI '90*, New York, New York, USA: ACM Press, 1990, pp. 249–256, ISBN: 0201509326. DOI: 10.1145/97243.97281.
- [36] J. Nielsen, “Usability inspection methods,” *Conference on Human Factors in Computing Systems - Proceedings*, vol. 1994-April, pp. 413–414, Apr. 1994. DOI: 10.1145/259963.260531. [Online]. Available: <https://dl.acm.org/doi/10.1145/259963.260531>.
- [37] L.-O. Bligård, “Utvecklingsprocessen ur ett människa-maskinperspektiv,” *Chalmers*, 2015. DOI: 10.13140/RG.2.1.1954.4400. [Online]. Available: www.acd3.se.
- [38] L. O. Bligård and A. L. Osvalder, “Predictive use error analysis Development of AEA, SHERPA and PHEA to better predict, identify and present use errors,” *International Journal of Industrial Ergonomics*, vol. 44, no. 1, pp. 153–170, Jan. 2014, ISSN: 0169-8141. DOI: 10.1016/J.ERGON.2013.11.006.
- [39] B. Gaver and J. Bowers, “Annotated portfolios,” in *Interactions*, vol. 19, New York, NY, USA: ACM, Jul. 2012, pp. 40–49. DOI: 10.1145/2212877.2212889. [Online]. Available: <https://dl.acm.org/doi/10.1145/2212877.2212889>.
- [40] A. Cooper, R. Reimann, D. Cronin and C. Noessel, *About face : the essentials of interaction design*, 4th ed. Indianapolis, IN: John Wiley & Sons, 2014.

- [41] *The Online Collaborative Whiteboard for Teams*. [Online]. Available: <https://www.figma.com/figjam/>.
- [42] Google, *Material Design*. [Online]. Available: <https://m3.material.io/>.
- [43] Microsoft, *Inclusive 101 Guidebook*, 2016.
- [44] Luftfartsverket, *Behörighetskrav*. [Online]. Available: <https://www.lfv.se/bli-flygledare/behorighetskrav>.
- [45] Luftfartsverket, *Så går urvalsprocessen till*. [Online]. Available: <https://www.lfv.se/bli-flygledare/tester>.
- [46] Microsoft, *Inclusive Design for Cognitive Exclusion Guidebook*, 2023.

A

Project Plan

This section contains the project time plan and a Gantt chart (figure A.1) visualising our intended working process. The dates in the Gantt chart are of the year 2024.

Research

- Interviews with Fredrik Falkman (head of innovation at SSRS) to familiarise with SSRS drone project.
- Literature review of articles within HCI, on HDI and Cognitive User Aspects of Multi-Control of autonomous drones, and robots.
- Understand cognitive user aspects through literature review.
- Interviews and site-visit to understand air traffic control tower through semi-structured interviews (4.1.3), and observations (4.1.4).
- Collection of other drone piloting interfaces and analysis of the graphical interface patterns to benchmark (4.1.2) existing work. (SQ1 and SQ2)

Analysis

- Evaluate air traffic control tower by sorting insights from site-visit through affinity diagramming (4.2.1).
- Evaluate other drone interfaces by sorting insights from benchmark through affinity diagramming (4.2.1).
- Frame research insights from literature review, and evaluations of air traffic control tower, and drone interfaces through affinity diagramming (4.2.1).
- Define problem space by pinpointing the research insights and translating them to user needs in a table of requirements. (SQ3)

Synthesis

- Ideation; Formulate initial ideas of what a drone GUI for SSRS could look like using brainstorming techniques 4.3.1.
- Conducting a co-design workshop 4.3.2 with participants from the user group and interaction design practitioners. A way to gain further insights from users and initiate design concepts.
- Build low fidelity concepts using user flow diagram 4.3.4 and wireframing 4.3.5.
- Evaluate conceptual prototypes through internal evaluations such as HTA, PHEA 4.3.8, and heuristic evaluation 4.3.7 (SQ2)

Realisation

- Creation of high fidelity prototypes in Figma to use for usability tests, to find answers to RQ
- Evaluate the final prototype through usability tests 4.4.1 with users (SQ2)
- Formulate findings of the usability tests (RQ)
- Visualise results by annotating essential parts of the interface to a concluded annotated portfolio 4.4.3.

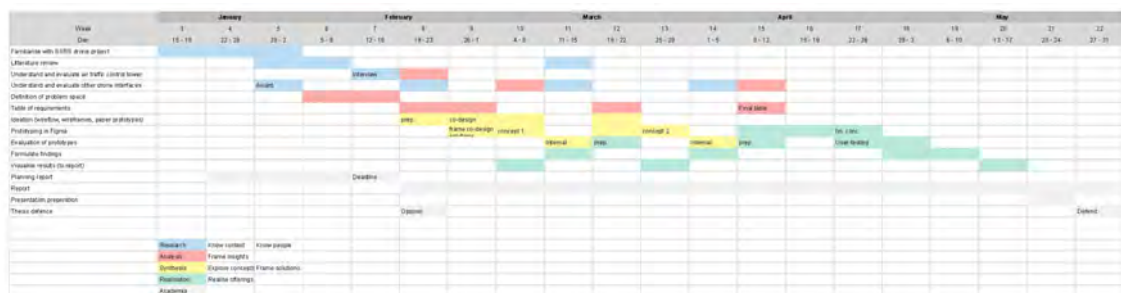


Figure A.1: Gantt schedule over thesis project plan

B

Identified Design Patterns from Benchmarking

B. Identified Design Patterns from Benchmarking

Feature	UgCS	QGC	Flytbase
Map		As background	Flat map and gps photo
Camera view		Map/camera-switch (acc. to youtube)	
Click and drag waypoints			
Video stream			
Settings pre-launch to enable different use cases			
Selection survey/corridor scan			
Customizable settings			
Listview (of drones with parameters)	In commander		
Multiple users on the server			
Artificial horizon			
Multi commands			
Mark and save obstacles			
3D vision (topography)			
Compressed to fit 1 monitor			
Altitude schematics			
List of waypoints			
Simultaneous video feed		Not seen	
Weather conditions			
Typing in aimed coordinates			
Marked out airspace/protected areas/caution area			
Automatic route planning			

C

Interview Template

Marcus, Aviant

Hur ser förberedelserna ut inför flygningen? Vad behöver du tänka på? Hur ställer du in detta i gränssnittet?

Hur ser hela flygningen ut, från förberedelser till landning?

Hur ser nivån av autonomitet ut? När behöver du hålla fokus? Finns det tillfällen då du inte alls behöver hålla koll?

I vilka situationer upplever du mest stress under en hel flygningsprocess? Varför?

Vad behöver du för att känna att du har kontroll när det kommer till gränssnittet? Varför?

Hur lång tid/hur många flygningar tog det att känna sig bekväm som pilot? Varför?

Hur många drönare kan du hantera samtidigt? Vad tror du är smärtgränsen? Vad blir skillnaden för dig ju fler drönare du behöver styra samtidigt? Blir det någon skillnad i gränssnittets layout?

Hur långt ifrån kan drönarna vara som en pilot styr samtidigt?

Kan flera piloter vara inne i gränssnittet samtidigt och styra separata drönare? Hur ser det då ut?

Finns det något som du upptäckt som varit extra smidigt eller tycker om lite extra, smått som stort? *Ibland i en app kan ett simpelt knapptryck vara något som användaren tycker är supernice av någon anledning.*

Vad för farhågor kan dyka upp under en flygning? Hur larmas detta i systemet?

Vilka parametrar är viktigast att alltid ha synligt i gränssnittet?

Finns det något som saknas i systemet?

Du pratade under flygningen vi var med på om emergencypunkter. Hur sätts dem ut? Gör du det eller gör QGC det?

Rickard, Landvetter

Vad är er uppgift?

Vad ser ni framför er när ni jobbar?

Vilka interaktioner är fysiska komponenter? Är det viktigt att de är just fysiska eller kan de göras digitala?

Vad görs manuellt i ditt arbete och vad sker automatiskt/av någon annan?

Hur viktigt är det att kunna se ut genom fönster? Vad kollar ni på då? Finns flygledartorn med dålig/låg/ingen visuell uppsikt? Hur gör dom då?

Styr man någonsin om flygplan till annan ort? Hur går det till då? Lämnas arbetet över?

Såg en video med två personer som besvarade vad som krävs för att bli en bra flygledare. Bra på teamwork och snabb på att ställa om var några av orden som kom upp. Vad tror du de menar med det? Vad tänker du på där dessa egenskaper är viktiga i flygledarrollen?

Sitter man någon gång ensam i tornet? Varför/varför inte?

Finns det något i systemet som du nästan aldrig kollar på (överflödigt)?

Finns det något som du önskar visades i systemet som du nu håller i huvudet?

Har alla flygtorn samma gränssnitt? Finns en standard? Hade du kunnat sätta dig i t ex Malmö och jobba där på studs?

Vad tror du är skillnaden på att övervaka flygplan och drönare?

Har du flugit drönare? Vilken typ? Hur skulle du jämföra flygledarjobbet med drönarflygning?

Vad har ni för omvärldsinformation (allt som händer runtomkring som kan påverka) för att kunna genomföra jobbet och för att känna att man har kontroll?

Hur mycket litar du på era system? Finns det situationer som du känner att du inte kan lita helt? Har det uppstått någon situation när systemet visat fel?

När de har många flyg/mycket att göra vad behövs för att känna att man har kontroll?

Får man som flygledare träning i stresshantering (alltså kortvarig stress, "jakt")?

Om en nödsituation skulle uppstå med ett plan, hur får du reda på detta? Vad händer med de övriga planen du blivit tilldelad?

Hur många flygplan håller folk koll på samtidigt och vad är smärtgränsen? Vad blir skillnaden för dig ju fler flygplan du behöver styra samtidigt? Blir det någon skillnad i gränssnittets layout?

Övriga tankar om hur flygledarrollen kan inspirera en drönarpilots arbete?

Magnus, Einride

Vad ser ni på era skärmar?

Vad behöver man förbereda innan en körning?

Vilka frågor behöver besvaras?

Vad är er uppgift?

Är ni ute och kör kommersiellt?

Hur ser förarsätet ut nu?

Just nu 1:1 förhållande, tittar ni på utökat förhållande?

Vad ser ni framför er när ni jobbar?

Hur många skärmar använder ni? Hur stor ungefär och vad syns på varje skärm? Hade det gått med bara en skärm tror du?

Vilka delar interagerar operatören med? Kontakt med andra personer/allt som finns på skrivbordet/

Vilka interaktioner är fysiska komponenter? Är det viktigt att de är just fysiska eller kan de göras digitala?

Vad görs manuellt i ditt arbete och vad sker automatiskt/av någon annan? (hur autonomt är det)

Styr man någonsin fordon på annan ort?

C. Interview Template

Sitter man någon gång ensam i styrplatsen? Varför/varför inte?

Finns det något i systemet som du nästan aldrig kollar på (överflödigt)?

Finns det något som du önskar visades i systemet som du nu håller i huvudet?

Litar du på systemet?

Har du flugit drönare? Vilken typ? Hur skulle du jämföra ditt jobb/förarjobbet med drönarflygning (autopilot)?

Vad tror du är skillnaden på att övervaka lastbilar och drönare?

Vad har ni för omvärldsinformation (allt som händer runtomkring som kan påverka) för att kunna genomföra jobbet och för att känna att man har kontroll? Hur får ni informationen?

Hur mycket litar du på era system? Finns det situationer som du känner att du inte kan lita helt? Har det uppstått någon situation när systemet visat fel?

Vad tror du behövs för att styra flera fordon samtidigt och fortfarande känna kontroll över situationen?

Om flera drönare får problem samtidigt, hur prioriterar man det?

Om en nödsituation skulle uppstå med fordonet, hur får du reda på detta i systemet?

Hur är skillnaden att styra ett fordon på olika geografiska avstånd?

Övriga tankar om hur din roll kan inspirera en drönarpilots arbete?

D

PUEA Table

D. PUEA Table

#	Error	Type	Cause	Primary consequence	Secondary consequence	Detection	Recovery	Protection (plan b)	Prevention
3.1	User types in wrong coordinate when trying to find the areas target	T2: Wrong message transmitted	Slip: Misheard message or finger slip	Drone will not arrive to correct target	5: Crew will not receive correct or proper footage	1: No detection before secondary cause	Add or move waypoint during operation	None	User receives coordinates in text, ability to copy and paste
3.2.1 a	Miscalculate drone time, when selecting drone	P3: Correct but inappropriate plan executed	Slip: User not calculating correctly	Inappropriate selection of drone	4: Loss of drone	5: Notice insufficient parameters during flight	Conduct controlled landing	"GPS"-tracker in drone, back-up drones available	Teach user during pre-training how to calculate and estimate properly
b	Misinterpreting the information displayed if drone data when selecting drone	C4: Wrong check on right object	Knowledge based: Unclear information; not knowing what one is looking at	Inappropriate selection of drone	4: Loss of drone	5: Notice insufficient parameters during flight	Conduct controlled landing	"GPS"-tracker in drone, back-up drones available	Make sure clear UI elements are used for easy understanding of information
c	User does not find drone data when determining drone selection	P4: Correct plan executed but too late	Knowledge based: Not knowing how to retrieve information	Delayed drone selection	5: Crew will not receive footage on time	5: Once trying to select drone the hovering will revise info	Speed up the rest of planning process	No protection	Information always visible
3.3.1 a	Misinterpreting map information displayed if drone data when selecting drone	R2: Wrong information obtained	Knowledge based: Map layers are not enough information for user to understand	Inappropriate action is being made	2: User falls with avoiding drone collision (in restricted area)	3: May be detected during flight	Double check with course material, adjust route	No protection	Make sure maps are clearly visualised
b	Making important information less so when using map layers	P1: Plan preconditions ignored	Violation: User is supposed to check map layers	Inappropriate action is being made	2: User falls with avoiding drone collision (in restricted area)	3: May be detected during flight	Adjust route	No protection	Nudging the use of map layers
3.3.2	User adding inappropriate amount of waypoints	P3: Correct but inappropriate plan executed	Knowledge based: User does not know how to perform the task	Drone will be delayed to site due to inefficient route	5: Crew will not receive footage on time	2: Difficult to detect before secondary consequence	Delete/add waypoints	None	Drone flying course. System giving initial suggestion.
3.3.3 a	User drone waypoints, inappropriate (overhead)	P3: Correct but inappropriate plan executed	Knowledge based: User does not know how estimate best placement	Drone will be delayed to site	5: Crew will not receive footage on time	2: Difficult to detect before secondary consequence	Adjust waypoints	None	Drone flying course. System giving initial suggestion.
b	User drone waypoints, inappropriate (in restricted area)	P1: Plan preconditions ignored	Violation: User doesn't follow the rules	Drone flies through restricted area	3: Drone might crash into other object	5: User know that error has been made	Adjust waypoints	None	User must learn why. System support not enabling WP in restricted area (RA) make it difficult to add in RA
4.1 a	Miscalculate drone time, before drone launch	P3: Correct but inappropriate plan executed	Slip: User not calculating correctly	Drone will be delayed to site due to inefficient route	4: Loss of drone	5: Notice insufficient parameters during flight	Conduct controlled landing	"GPS"-tracker in drone, back-up drones available	Teach user during pre-training how to calculate and estimate properly
4.1 b	Misinterpreting of drone data before launching drone	C4: Wrong check on right object	Knowledge based: Unclear information; not knowing what one is looking at	Drone will be delayed to site due to inefficient route	4: Loss of drone	5: Notice insufficient parameters during flight	Conduct controlled landing	"GPS"-tracker in drone, back-up drones available	Make sure clear UI elements are used for easy understanding of information
4.1 c	User launches drone before that the route plan	P3: Correct but inappropriate plan executed	Slip: User accidentally clicked on button	Drone will be delayed to site due to inefficient route	3: Drone might crash into other object	5: Interface changes to flying the drone	Exit operation	None	Make interaction uneasy
Determining when focus can be put elsewhere									
11.8.2	User fails to notice nearby objects by misinterpreting information, being too focused on terrain, or distance	C2: Checking incomplete or A4: Action too little	Lapse: User forgets to get an appropriate overview	User misses nearby objects, drone continue on route	2: Fails to avoid (risk of) collision	3: May be detected if user gets reminded	Adjust route	None	System reminds/ guides user to check transponder map and camera
2.3	User fails to make the drone enter other way means to stay out of the way from other prioritised objects	A8: Action omitted	Knowledge based: User does not know how to	Drone will not "pause" its route	2: User cannot direct focus on prioritised drones, fails to avoid collision	5: User will know that drone will not "pause"	None	User conducts a controlled landing instead	Drone flying course, system workshop
	User not checking the whole camera view	C2: Checking incomplete or A4: Action too little	Lapse: User forgets the whole environment	User missing important information	4: Drone takes damage/loss of drone	3: May be detected if user gets reminded	User double check camera view, can take right decisions	"GPS"-tracker in drone, back-up drones available	System reminds/ guides user to check camera view
	User not keeping proper track of drone values thus expecting wrong values	C2: Check incomplete	Lapse: User forgets to keep track of values	User losing control of drone	4: Drone takes damage/loss of drone	2: Detected if user double check status	Double checking values and take other conclusions	"GPS"-tracker in drone, back-up drones available	System reminds/ guides user to check values
2.12.3	User misinterpreting support status	C4: Wrong check on right object	Knowledge based: User not understanding parameters	User takes conclusions based on wrong information	5: Crew will not receive footage on time	2: Detected if user double check status	User double check parameters and change plan	None	Clear units, icons
2.12.4 a	User starts when drone when task is clear (suspecting attention needed)	P2: Incorrect plan executed	Knowledge based: User miscalculated the situation to direct focus elsewhere	Drone delayed	5: Crew will not receive footage on time	3: Short time span of detection before primary consequence	None	None	System double checking "Are you sure, this will delay, is track clear?"
Finishing flight									
7.2	User lands drone on inappropriate spot	P4: Correct plan but inappropriate plan executed	Slip: User inattentive	Drone landing too far or too close to crew	4: Loss of drone	4: User is likely to detect the error by checking camera view when landing	User discontinues downwards motion, and finish the landing in other spot	None	Well defined required distance to human, drone flying course
#	Error	Type	Cause	Primary consequence	Secondary consequence	Detection	Recovery	Protection (plan b)	Prevention

E

Consent Form

Consent and information about processing of personal data in student thesis

I agree to my personal data in the form of:

- First name
- Family name
- Occupation/the role in which you are interviewed
- Telephone number
- E-mail address
- Audio recordings
- Video recordings
- Pictures

may be treated by Chalmers University of Technology for the study:

UI for Remote Operation of Drones: Designing a user interface for remote operation of autonomous, fixed-winged drones to support sea rescue missions. Study supported by user feedback, such as interviews, user evaluations, and questionnaires.

Information

Your personal data will be handled as follows:

May be used in report:

- First name
- Family name
- Occupation/role
- Pictures
- Quotes from interview

Stored information:

- E-mail address, telephone number will be saved until the end of the project, in case we want to contact you about follow-up questions.
- Audio and video recordings will be used for transcript and observations, and will be deleted after the project is finished

The data will be stored in a One-drive folder (with GDPR certification). The data outside of the ones used in the final report will be deleted after the end of the project or until consent is withdrawn. The final report will be published on Chalmers ODR and accessible to the public.

Your consent is valid until further notice. You have the right to withdraw your consent at any time. You do this through contacting stinaol@chalmers.se , alihol@chalmers.se or registrator@chalmers.se

If you withdraw your consent, we will cease processing personal data we have collected with the support of your consent. Some information may be saved due to Chalmers obligations under Swedish archive legislation.

Chalmers University of Technology, org. No. 556479-5598 is

personal data controller. You can find Chalmers [privacy policy](#) at www.chalmers.se.

As a participant you have the right to receive information about how your personal data is processed. You have the right to have incorrect information corrected, redundant data deleted, request that processing shall be restricted and data transferred to another actor. You also have the right to submit a complaint to the Swedish Authority for Privacy Protection (Integritetsskyddsmyndigheten). Do you have any questions about Chalmers's processing of personal data contact Chalmers's data protection officer at dataskydd@chalmers.se.

I agree that Chalmers University of Technology processes personal data about me in accordance with the above.

Place:	Signature
Date:	Name clarification

The form is drawn up in duplicate.

F

Matrices of Results from PUEA

F. Matrices of Results from PUEA

Table 5.11 Matrices for presenting the results of the PUEA analysis

Name	Content	Explanation
Matrix A:	Consequence versus task	Shows the tasks with the most serious error consequences
Matrix B:	Error type versus task number	Shows which type of use error exists in the various tasks
Matrix C:	Error cause versus task number	Shows the causes of the use errors in the different tasks
Matrix D:	Error type versus secondary	Shows which error type gives rise to the highest risks
Matrix E:	Error cause versus secondary	Shows which error cause gives rise to the highest risks
Matrix F:	Error cause versus error type	Shows the coupling between error cause and error type
Matrix G:	Detection versus task number	Shows in which tasks errors are difficult to detect
Matrix H:	Detection versus error type	Shows which type of error is difficult to detect
Matrix I:	Detection versus error cause	Shows the causes of errors that are difficult to detect
Matrix J:	Detection versus secondary	Shows severity of consequences for errors that are difficult

Matrix A

Consequence/Task	1	2	3	4	5	6	7
1	-	-	0	0	0	-	0
2	-	-	2	0	2	-	0
3	-	-	1	0	0	-	0
4	-	-	2	2	2	-	1
5	-	-	4	0	2	-	0

- Allvarligast konsekvenser när ruten planeras fel
- Allvarliga konsekvenser om drönanaren monitoreras fel
- Milda konsekvenser om drönanaren inte avfyras eller landar fel

Matrix B

Error type/Task	1	2	3	4	5	6	7	Tot
Planning	-	-	6	1	1	-	1	9
Action	-	-	0	0	1	-	0	1
Checking	-	-	1	1	4	-	0	6
Retrieval	-	-	1	0	0	-	0	1
Communication (T)	-	-	1	0	0	-	0	1
Selection	-	-	0	0	0	-	0	0

- Vanligaste typen av fel är felbedömning av planering
- Operatören kan missa checkar under flygning

- Under planering av rutt är felplanering det vanligaste användar-felet.
- Under flygning är checking det vanligaste användar-felet.

F. Matrices of Results from PUEA

Matrix C

Error cause/Task	1	2	3	4	5	6	7	Tot
Lapse	-	-	0	0	3	-	0	3
Slip	-	-	2	1	0	-	1	4
Rule-based	-	-	0	0	0	-	0	0
Knowledge-based	-	-	5	1	3	-	0	9
Violation	-	-	2	0	0	-	0	2

• Vanligaste orsaken är att operatören inte vet bättre
• Operatören missar delar eller glömmer bort varför det är viktigt att genomföra steget

Viktigt att utbilda

• Under planering av rutt är det viktigt att piloten vet hur man ska göra, vilka regler som gäller och varför.
• Under flygning är det viktigt att piloten vet hur man använder systemet samt att hen blir påmind genom gränssnittet.

Matrix G

Detection/Task	1	2	3	4	5	6	7
1	-	-	1	0	0	-	0
2	-	-	2	0	2	-	0
3	-	-	2	0	3	-	0
4	-	-	0	0	0	-	1
5	-	-	4	2	1	-	0

• Viktigt att upptäcka felet vid planering, annars påverkar det vidare delar av flygningen, alternativt kan piloten inte avfylla drömlaren och hela flygoperationen blir inställd
• Under flygning kan piloten upptäcka felet om hen är uppmärksam under flygningen, kräver stort fokus, systemet bör vara transparent och hjälpsamt

• Lätt att upptäcka fel både vid avfyllning och landning

G

User Evaluation Reference Guide

Nu startar ditt jour-pass som drönpilot. Det är en morgon i Augusti och det väntas vara många ute på sjön idag, så var beredd på att larmcentralen kommer förfråga flera drönare i olika delar av landet.

Cheatsheet

ETA: Estimated time of arrival

ETS: Estimated time at scene

WP: Waypoint

POI: Point of Interest

Drönaren vill starta i motvind.

Allt som oftast vill man att drönaren ska komma fram så snabbt som möjligt, även om detta tar mer på batteriet.

Drönaren landar i vattnet när operationen är klar.

Flyg ej i restriktionområden (röda områden)

Undvik att flyga över bebodda öar

Undvik att flyga över båtar

Undvik höga landmärken och lågt flygande flygplan.

Du kan väja på 4 sätt:

- Landa
- Sväng skarpt höger
- Loitra (cirkulera på plats)
- Flytta på Waypoint

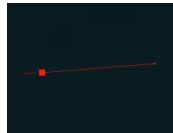
Andra saker du kan göra:

- Välja passande drönare
- Korrigera rutt eller tid vid behov
- Starta rutt
- Övervaka omgivning och parametrar

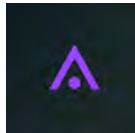
Då detta är en prototyp finns begränsningar till vad som går att göra. Det är därför bra om du tänker högt innan du agerar i gränssnittet, vad du rent instinktivt hade velat göra innan du gör det. Ifall du upptäcker något fel i gränssnittet så berätta det för oss.



1. Sjöfarkost



2. Luftfarkost



3. Statiskt höjdhinder

Om du har frågor om specifik info får du gärna fråga, ibland svarar vi.

H

Answering Time – Rapid Questions in User Evaluation

H. Answering Time – Rapid Questions in User Evaluation

Number of drones	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Mean time	(Number of drones)/(Answering time, s)
P1										
2	34	20	2						18.666667	
3	10	11	1	3					6.25	
4	6	5	2.5	11	3	1	8		5.2142857	
5	8	9	4	1	5	6	6	10	6.125	
P2										
2	42	25	1.5						22.833333	
3	19	8	1	4					8	
4	6	6	2	10	3	1	8		5.1428571	
5	6	10	4	1	4	3	8	6	5.25	
6	11	3	9	10					8.25	
P3										
2	15	1.7	1						5.9	
3	6	2	0	2					2.5	
4	12	2.5	1	9	5	2	4		5.0714286	
5	3	2	2	0.5	3.5	0.7	0.5	1.4	1.7	
6	1.2	1	0.5	4.5	3	4	1	2.7	2.2375	
7	8.9	1		4	2	6.8			4.54	
P4										
2	15	5	2						7.3333333	
3	2	12	1	3					4.5	
4	1	1	1	1	3	2	4		1.8571429	
5	1.5	2	2.5	1	1	1	1	2	1.5	
6	1.5	2	3	2	2	1	1.5		1.8571429	
7	10	1	1.5	1.5	3	2			3.1666667	

