

Increasing human supervision over operational assets for persistent and efficient ocean observation

Afonso Carvalho Pereira de Sá



Mestrado em Engenharia Informática e Computação

Supervisor: João Tasso de Figueiredo Borges de Sousa

Supervisor: António Sérgio Borges dos Santos Ferreira

July 31, 2022

Increasing human supervision over operational assets for persistent and efficient ocean observation

Afonso Carvalho Pereira de Sá

Mestrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

President: Prof. Jácome Miguel Costa da Cunha

Referee: Prof. Lino Forte Marques

Referee: Prof. João Tasso de Figueiredo Borges de Sousa

July 31, 2022

Abstract

There are several entities at the forefront of combining ocean exploration and monitoring with robotics; among them is the LSTS (Laboratório de Sistemas e Tecnologia Subaquática). The LSTS developed Ripples as a communications hub for data dissemination, pre-planning, and situation awareness. This platform aims to collect and centralize data from different sources and maintain a synoptic operational view, enabling, for example, coordination among geographically distributed teams. This software has been used in several maritime campaigns, where it is necessary to make critical decisions. These decisions must be informed; as such, operators must have access to all necessary information.

With Ripples' growth, it became less intuitive and cluttered. New functionalities and datasets were added without much concern to the user interface and experience, which proved detrimental to maintaining situational awareness and navigating the data being displayed.

This dissertation was proposed to mitigate the issues with Ripples' interface. As such, an overhaul was performed. The objective was to use design and human-computer interaction techniques, such as Ecological Interface Design (EID), to build the new interface, with the intent of reducing visual entropy and improving planning and supervision of missions in remotely controlled environments.

In order to achieve those goals, it was necessary to study design practices, similar platforms, maritime UI, and Ecological Interface Design. During this phase, methods of measuring the workload and situational awareness values of an operator were studied to evaluate the interface's performance. The research process led to an informed implementation phase in which the principles of EID and design practices were applied. This phase led to the completion of the redesigned interface. Afterwards, it was necessary to test the old and new interfaces, by measuring workload and situational awareness values with ten selected operators, and compare their performances. From the results, it was possible to conclude that all the objectives were met, with the new interface reducing workload by 31%, increasing situational awareness by 18%, reducing entropy between users, and increasing the information provided.

This dissertation is integrated with the efforts of the LSTS and the Norwegian University of Science and Technology (NTNU) to control and supervise operation networks of autonomous and crewed vehicles. It is also relevant to the work being made with the LSTS' partners (NATO-CMRE, NTNU, NASA-AMES) to implement data centralization.

Keywords: Human-computer Interaction; Situational Awareness; Workload; UI/UX; Ecological Interface Design;

Resumo

Existem várias entidades na vanguarda de combinar a robótica à exploração e monitorização dos oceanos, sendo uma delas o Laboratório de Sistemas e Tecnologia Subaquática. O LSTS desenvolveu o Ripples como um meio de comunicação e disseminação de informação, pré-planeamento e consciência situacional. Esta plataforma tem como objetivo colecionar e centralizar informação de diferentes fontes de modo a manter uma situação global do ambiente, permitindo, por exemplo, a coordenação de equipas geograficamente separadas. Este software tem sido utilizado como ferramenta de apoio em campanhas marítimas, onde existe a necessidade de tomar decisões críticas. Estas decisões têm de ser informadas, como tal é necessário que os operadores tenham acesso a toda a informação necessária.

Com os desenvolvimentos do Ripples a sua interface ficou menos intuitiva e mais desordenada. A adição de novas funcionalidades, datasets e utilizadores foi feita sem grande cuidado relativo ao design da interface e da experiência do utilizador. Isto provou ser prejudicial para manter a consciência situacional e aumentou a carga de trabalho ao navegar a informação contida no Ripples.

Esta tese foi proposta com o intuito de mitigar os problemas da interface do Ripples, de tal modo, foi proposto rever esta interface; o objetivo desta revisão é aplicar técnicas de design e de interação pessoa-computador, por exemplo o Ecological Interface Design, no novo design do Ripples de modo a remover entropia visual e aumentar a capacidade de supervisão em missões de ambiente remoto.

De modo a cumprir esses objetivos foi necessário estudar técnicas de design, plataformas similares, interfaces marítimas e o Ecological Interface Design. Ao longo deste período estudou-se também métodos para medir a carga de trabalho e consciência situacional de um operador, de modo a avaliar a performance da interface. O processo de pesquisa levou à fase de implementação onde se aplicaram os princípios de EID e design. Esta fase culminou com a nova interface. Dado isto, foi necessário testar e comparar a performance da interface antiga e a nova. Uma vez obtidos os resultados foi possível concluir que todos os objetivos foram cumpridos, o valor da carga de trabalho reduziu em 31%, a consciência situacional aumentou 18%, reduziu-se a entropia entre diferentes utilizadores e aumentou-se a quantidade e qualidade de informação disponível.

Esta dissertação está enquadrada nos desenvolvimentos em controlo e supervisão de redes operacionais de veículos autónomos e tripulados, partilhados entre o LSTS e a Norwegian University of Science and Technology (NTNU). Paralelamente, este trabalho também se insere nos esforços do LSTS e dos seus parceiros (NATO-CMRE, NTNU, NASA-AMES) para a implementação e centralização de dados em ambientes remotos de operação.

Palavras-chave: Interação Pessoa-Computador; Consciência Situacional; Carga de Trabalho; UI/UX; Ecological Interface Design;

Acknowledgements

First, I would like to thank my supervisors, Professor João Sousa, for providing me with the opportunity of working on this project and for the feedback that was given, and Sérgio Ferreira for his supervision, all the help during the semester, and for being a guiding hand that helped shape this document. To Renato Mendes, thank you for being a welcoming presence in the laboratory, for all the help and feedback during the semester, and for giving me the opportunity to present my work at "Encontro de Oceanógrafos 2022".

To all the team at the LSTS, I must thank you for all the support and help, all the entertainment, and the lighthearted environment that helped me achieve the goals we set for this project.

To my friends, I must thank you for all the unwavering support and words of encouragement when I most needed them; I must also thank you for the help you provided whenever I needed it. Most importantly, I want to thank you for being an endless source of entertainment and great company. I'm confident this project would not have been as successful without you.

Lastly, I want to thank my parents for providing me with the best education, the opportunities you have given me, and all the patience and emotional support. The efforts you put into educating me and providing me with all the opportunities to succeed have gotten me to this point. To my brother, thank you for being the best role model I could have asked for. Thank you for breaking the ground I tread; following in your steps has been a pleasurable experience. You are headed for greatness, and I hope I can follow you along this journey.

Afonso Carvalho Pereira de Sá

‘There is no perfection only life’

Milan Kundera, *The Unbearable Lightness of Being*

Contents

1	Introduction	1
1.1	Underwater Systems and Technology Laboratory	2
1.2	REP(MUS)	2
1.3	Ripples	3
1.4	Motivation	4
1.5	Objectives	5
1.6	Document Structure	5
2	Related Work	6
2.1	Similar Platforms	6
2.1.1	ERMA	6
2.1.2	CMEMS MyOcean Viewer	7
2.1.3	ODSS	8
2.2	UI/UX	8
2.2.1	Design practices	8
2.2.2	Maritime UI	9
2.2.3	Ecological Interface Design	10
2.3	Situational Awareness	11
2.3.1	SAGAT	11
2.3.2	SART	12
2.4	Workload	12
2.4.1	NASA-TLX	12
2.4.2	SWAT	13
2.5	Performance Metrics Summary	14
3	Initial Approach	15
3.1	Analysis of the requirements - User Questionnaire	15
3.2	Analysis of the system	17
3.3	Exploring the interface and initial mock-ups	19
4	Implementation	21
4.1	Ecological Interface Design	21
4.1.1	AIS Ship highlight	21
4.1.2	Past and future prediction feedback	22
4.1.3	Winding number algorithm	24
4.1.4	Layer legend consistency	26
4.1.5	Role-specific view	28
4.2	New sidebar design	31

4.2.1	Layer grouping	32
4.2.2	Custom labels	33
4.3	Plan feedback	34
4.3.1	High density areas	35
4.3.2	AIS Ship is in operating window	35
4.4	Layer scale definition	36
4.5	Simplified AIS information panel	37
5	Results and Discussion	39
5.1	Performance Metrics	39
5.2	Tests	39
5.2.1	Test Scenario	40
5.3	Results	43
5.3.1	Workload	43
5.3.2	Situational Awareness	45
5.4	Discussion	48
6	Conclusions	50
6.1	Summary of the Work Done	50
6.2	Objectives Completion	51
6.3	Future Work	52
	References	53
A	Users Questionnaire	56
B	NASA TLX	65
C	SART	67
D	Workload and SA scores	68
E	Presentation at Encontro de Oceanografia 2022	69

List of Figures

1.1	LSTS	2
1.2	LSTS at REP(MUS)	3
1.3	Ripples being used at REP(MUS)	3
1.4	Ripples - Future States	4
2.1	ERMA and Ripples showing similar information	7
	(a) ERMA - Vehicle Density	7
	(b) Ripples - AIS Density	7
2.2	CMEMS	7
	(a) CMEMS - One Layer	7
	(b) CMEMS - Two Layers	7
2.3	ODSS	8
2.4	MOCU - Vehicle and mission status	10
2.5	NASA TLX - App	13
3.1	Database Schema	18
3.2	Communication Diagram	18
3.3	Ripples mock-up for layer priority	19
3.4	Ripples mock-up for layer legends	20
3.5	Ripples - Sea Level Anomaly	20
4.1	AIS Ship - Old version	22
4.2	AIS Ship - New version	22
4.3	AIS Ship Prediction - Old version	23
4.4	AIS Ship Prediction - New version	23
4.5	AIS Ship - Trajectory and past positions	24
4.6	Dan Sunday's winding number algorithm	26
4.7	AIS Ship - Lines and predicted states (New algorithm)	26
	(a) Predicted state about to cross land	26
	(b) Predicted state does not cross land	26
4.8	Ripples - Legend area	27
4.9	Ripples - Legend example	27
4.10	Ripples - New legends	28
4.11	Ripples AIS - Scientists old view	29
4.12	Ripples AIS - Scientists new view	29
4.13	Ripples - Operator old view	30
4.14	Ripples - Operator new view	30
4.15	Custom LayersControl	32
4.16	Sidebar comparison (Old vs New)	33

4.17	Scale values and units - Hard to read	34
4.18	Scale values and units - New label design	34
4.19	Warning message on plan creation	35
4.20	Two plans - Warning operating window	36
4.21	Copernicus automatic scaling	36
4.22	Layer scale definition	37
	(a) Global velocity scale	37
	(b) Local velocity scale	37
4.23	AIS information panel - Old	38
4.24	AIS information panel - New	38
5.1	Reference plan	41
5.2	Locations to read the Sea Surface Temperature	41
5.3	Location to read the Sea Level Anomaly	41
5.4	Location to read the Wind intensity and direction	42
5.5	Location of resurfaced LAUV	42
5.6	NASA TLX - Scale weights old vs new	43
5.7	NASA TLX - Weighted scores old vs new	44
5.8	Individual workload scores	45
5.9	SA scores	46
5.10	Individual SA scores	48
B.1	NASA TLX - Scoring	65
B.2	NASA TLX - Weights	66
C.1	SART	67
D.1	Workload and SA Scores	68

List of Tables

5.1	Workload scores	45
5.2	Situational Awareness scores	47
5.3	Overall results	49

Listings

4.1	Winding number algorithm	25
4.2	Layer scale code	37

Abbreviations

AIS	Automatic Identification System
AUV	Autonomous Unmanned Vehicle
CSS	Cascading Style Sheets
GIS	Geographic Information System
HTML	HyperText Markup Language
LAUV	Light Autonomous Unmanned Vehicle
LSTS	Laboratório de Sistemas e Tecnologia Subaquática
NASA TLX	National Aeronautics and Space Administration Task Load Index
REP(MUS)	Robotic Experimentation and Prototyping Augmented by Maritime Unmanned Systems
SA	Situational Awareness
SAGAT	Situation Awareness Global Assessment Technique
SART	Situation Awareness Rating Technique
SPAM	Situation Present Assessment Method
SWAT	Subjective Workload Assessment Technique
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
UI	User Interface
USV	Unmanned Surface Vehicle
UX	User Experience

Chapter 1

Introduction

Context

There are several entities at the forefront of combining ocean exploration and monitoring with robotics; among them is the LSTS (Laboratório de Sistemas e Tecnologia Subaquática). This laboratory uses several types of autonomous vehicles to monitor and map the ocean to achieve persistent and efficient ocean observation. These vehicles, although autonomous, are not self-sufficient and need human supervision [15]. When deploying these vehicles, it is common to have multiple teams geographically separated, so they need to be able to control these vehicles and communicate efficiently.

One such tool used for this purpose is Ripples [27], a communications hub for data dissemination and situation awareness in operations at sea, including oceanographic campaigns using traditional and new robotic and autonomous observation systems. This software is integrated within the LSTS toolchain for command and control and complements it by improving the awareness of operators and scientists managing assets deployed in the open sea.

The LSTS cooperates with several outside organisations, resulting in highly collaborative maritime campaigns. One such example is the REP(MUS) [6, 25] 2021 exercise, a collaborative exercise involving more than 40 entities, including 17 Navies, 15 Research & Development Entities, one University, and eight organizations of NATO.

This dissertation is integrated within the LSTS and aims to improve Ripples' current interface by reducing visual clutter; increasing the quality and quantity of information provided; reducing workload; increasing situational awareness. It complements the current efforts of the laboratory to develop Ripples into being capable of Multisystem Command and Control (C2).

1.1 Underwater Systems and Technology Laboratory

The Underwater Systems and Technology Laboratory (LSTS) is a laboratory that was founded in 1997. It is composed of students, researchers, and professors from various engineering departments and Computer Science. The laboratory specialises in the design, construction, and operation of unmanned underwater, surface, and air vehicles, as well as developing the tools and technologies necessary to deploy networked vehicle systems. During its existence, researchers from the LSTS have conducted missions with their unmanned vehicles in the Atlantic and Pacific oceans and the Mediterranean sea.



Figure 1.1: LSTS

1.2 REP(MUS)

REP(MUS) is the largest robotics exercise in the world; it is co-organised by the Portuguese Navy and the LSTS. REP(MUS) is a multiple-day exercise with the participation of NATO and some of its members' Navies and other national and international organisations.

The vision with REP(MUS) is to advance the state of the art in networked vehicle systems through large-scale experimentation and cooperation involving Academia, Industry and the Operational Community. During this event, several unmanned autonomous vehicles are deployed, UAVs, AUVs, USVs, and UGVs, in order to test and demonstrate their uses and how they can complement manned teams (Figure 1.2). One of the goals set by the LSTS with REP(MUS) 2021 was to demonstrate Ripples' enhanced capabilities for marine situational awareness and multi-domain unmanned vehicle missions (Figure 1.3).

The laboratory intends to present an improved version of Ripples, having addressed its weaknesses for this year's event. The work done during this dissertation aims to help with this goal by addressing one of Ripples' weakest points, its interface.



Figure 1.2: LSTS at REP(MUS)

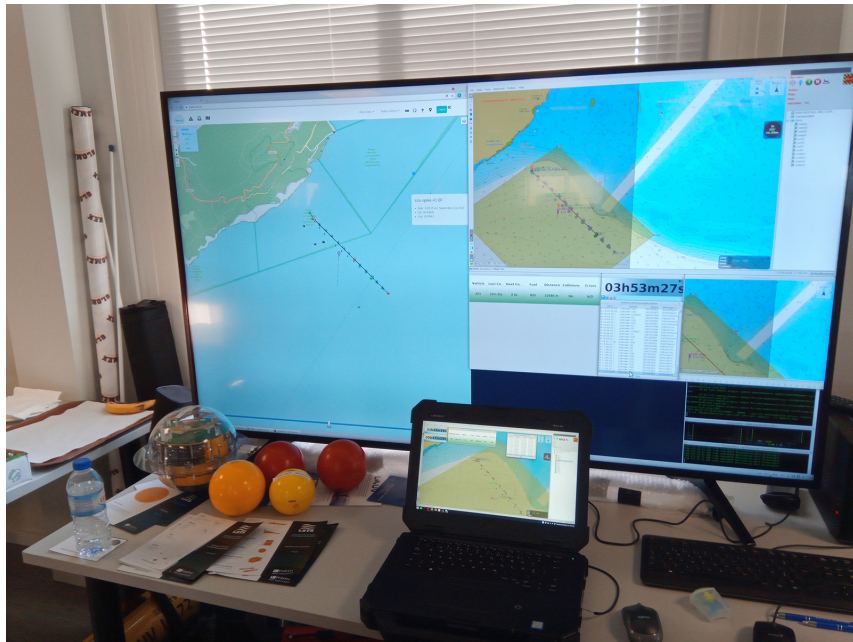


Figure 1.3: Ripples being used at REP(MUS)

1.3 Ripples

LSTS works with a toolchain composed of several tools that envelop the software embedded in the vehicles, a communication protocol, a distributed command and control infrastructure, and a “communications hub for data dissemination and situation awareness”. [13]

The cloud service, Ripples, is a secure, persistent, and collaborative maritime data processing & analysis service aimed at high-level decision-making and situational awareness. This module, Ripples, is a tool that aims to collect data from different sources and maintain a global situation, enabling coordination among geographically distributed teams.

It collects data from assets that have been deployed as well as other vehicles and is capable of querying their past states as well as planned future states (Figure 1.4). Ripples also provides data from weather forecasting services, bathymetry, and different maps, among other relevant layers.

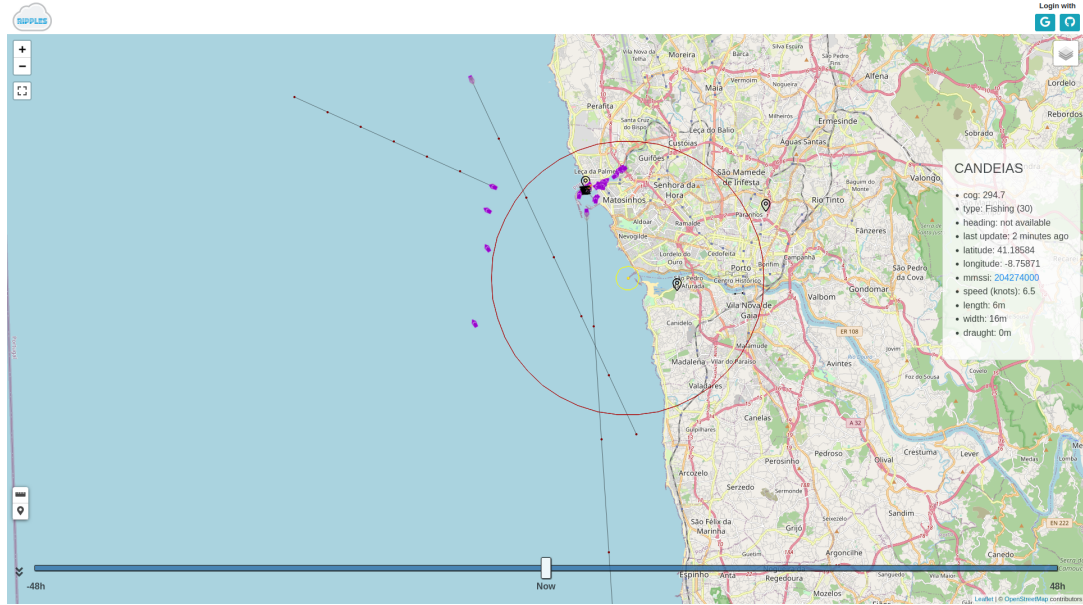


Figure 1.4: Ripples - Future States

1.4 Motivation

Ripples is used in remote environment campaigns where multiple autonomous vehicles are controlled simultaneously. The environment surrounding these vehicles "has a complex spatio-temporal variability and uncertainty".[22] As such, it is necessary to make critical decisions during the lifespan of a mission, for example conducting an asset's risk analysis [11, 35]. These decisions must be informed; Ripples acts as a platform that displays the necessary information to inform the operator's decisions. It is necessary that Ripples is capable of displaying the information clearly and that all crucial information is promptly available in order to reduce the operator's workload and increase their situational awareness.

This software is also used when planning missions by operators and scientists, such as oceanographers, to obtain information that is relevant during the planning process. It is important that Ripples displays the correct information and that it is easy to read and interpret. Moreover, different users have different requirements for Ripples. It is necessary to display different information for the different roles of users in order to reduce the entropy between them.

1.5 Objectives

The objectives for this dissertation are relatively straightforward, as it is the intent to improve Ripples' interface whilst keeping in mind the core functionalities. As such, the main objectives for the new version of Ripples are:

- **Reduce** entropy between users;
- **Maximise** displayed information;
- Maintain or **decrease** workload;
- Maintain or **increase** situational awareness;

1.6 Document Structure

The current chapter contextualises the work to be done during this dissertation, the LSTS, REP(MUS), the flagship maritime campaign, Ripples, and introduces the motivation and objectives for the work to be done.

The second chapter introduces the concept of situational awareness and workload and how they can be used to measure an interface's performance. It also describes the research done on the existing literature, containing a multitude of techniques to measure situational awareness and workload. Furthermore, this chapter presents the state of the art of UI design patterns and practices. It introduces some related work, user interfaces used in the maritime domain, and some design practices previously used in this area, such as Ecological Interface Design.

The third chapter describes the initial approach to the problem. It focuses on the analysis of the working domain of the system, a user questionnaire whose intent was to gather user requirements and feedback and contains some mock-ups of the interface.

Chapter four describes the implementation phase; it describes the work done and the rationale behind the design choices that were applied along this process.

The fifth chapter contains an explanation of how the performance metrics will be measured, what tests were done, and what results were obtained from those tests. There is also a statistical analysis to evaluate the statistical significance of the results.

In the final chapter, there is a brief summary of the work done during this thesis, a conclusion on the results of this work and the objective completion, and lastly, there is a description of future work to be done with Ripples in order to improve it further.

Chapter 2

Related Work

This chapter will introduce the concepts of workload and situational awareness, providing a definition of these concepts, multiple techniques to measure situational awareness and workload, and their pros and cons. There will also be a discussion of UI design patterns, designing human-computer interaction based on the decision-making process, and maritime UI. In this chapter, it is also possible to find similar software to Ripples and how these can aid the design process.

2.1 Similar Platforms

There are other platforms that allow the visualisation of weather, marine data, and maritime traffic, as well as platforms to help with decision-making and command and control of unmanned vehicles. The following are some of the most similar or relevant to Ripples.

2.1.1 ERMA

Environmental Response Management Application (ERMA) [19] is a web-based Geographic Information System (GIS) developed by the National Oceanic and Atmospheric Administration and the University of New Hampshire Coastal Response Research Center. It is intended for environmental responders and natural resource decision-makers to make informed decisions for environmental response, damage assessment, recovery, and restoration.

Much like Ripples, ERMA can display multiple layers with data spanning from environmental data, bathymetry, weather, and oceanography. It differentiates from Ripples by being more oriented to response planning and incident response. Its strength compared to Ripples is the quantity of layers and information that is provided by this service. However, this software suffers from some of the same issues as Ripples, such as difficulty in interpreting multiple layers and a hard to navigate sidebar. Although it is better organised than Ripples'. It has its own problems, such as a hard to navigate interface with relevant information that should be promptly visible, such as scales, hidden in different tabs in the sidebar.

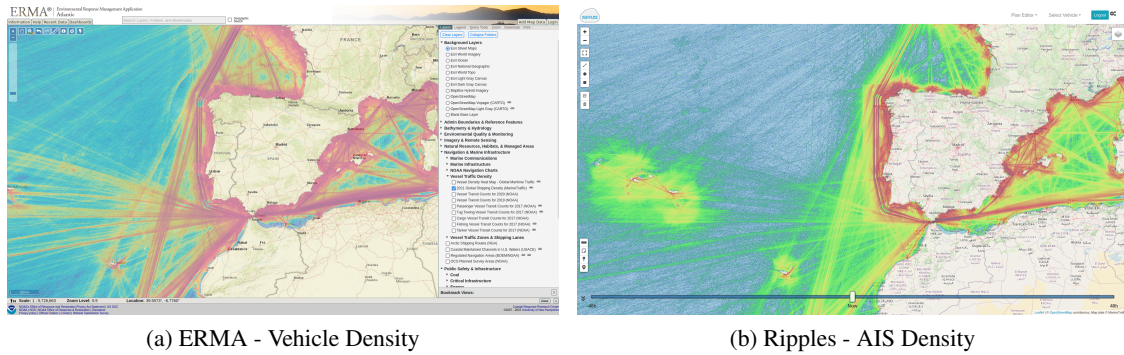


Figure 2.1: ERMA and Ripples showing similar information

2.1.2 CMEMS MyOcean Viewer

CMEMS MyOcean Viewer [31] is the richest data visualisation platform, that we found, in both data and how it is presented to the user. It serves as a catalogue for the Copernicus Marine Service and holds more than 300 layers which can be visualised in four dimensions: latitude, longitude, depth (height), and time. It is also possible to automatically generate graphs to display several variables on a single point and consult previous and forecast values.

This tool has a clear focus on data visualisation. It implements several quality-of-life features that would benefit Ripples, such as the possibility to establish layer's priorities by dragging them to their desired priority on the layer selection sidebar, the possibility to define the layer's opacity with a slider, or the possibility to change the layer's colour scale. Moreover, the UI of this website is very intuitive and organised.

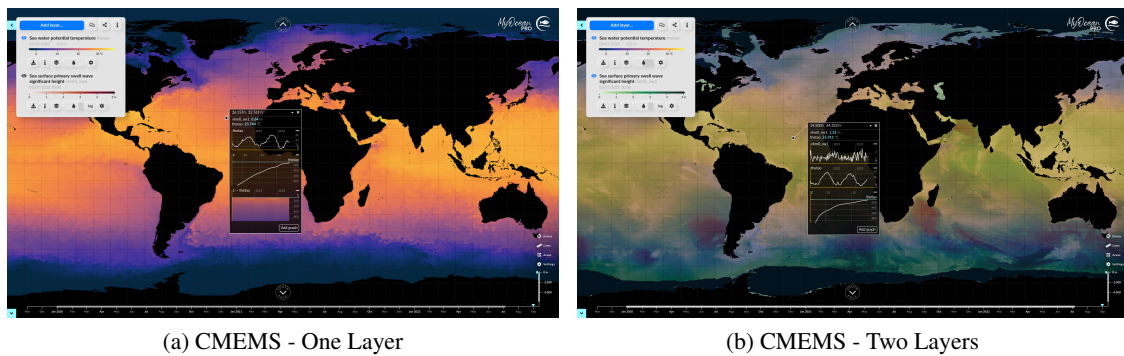


Figure 2.2: CMEMS

2.1.3 ODSS

Oceanographic Decision Support System (ODSS) is the platform that closely matches the idea of Ripples; it is a platform with the "[...] goal of improving our coordinated field campaign management and planning". [14]

ODSS and Ripples share the same background, intending to aid with planning campaigns and command and control of unmanned vehicles. They also share the capability to visualise and interpret data relevant to the decision-making process of said campaigns. It lacks in some features compared to Ripples and also shares some weaknesses. It is not possible to store plans and associate these trajectories to vehicles, it is also lacking in the quantity of layers and information that is possible to visualise. Moreover, this software shares the same interface problems as Ripples, which worsens the user experience when using this software (Figure 2.3).

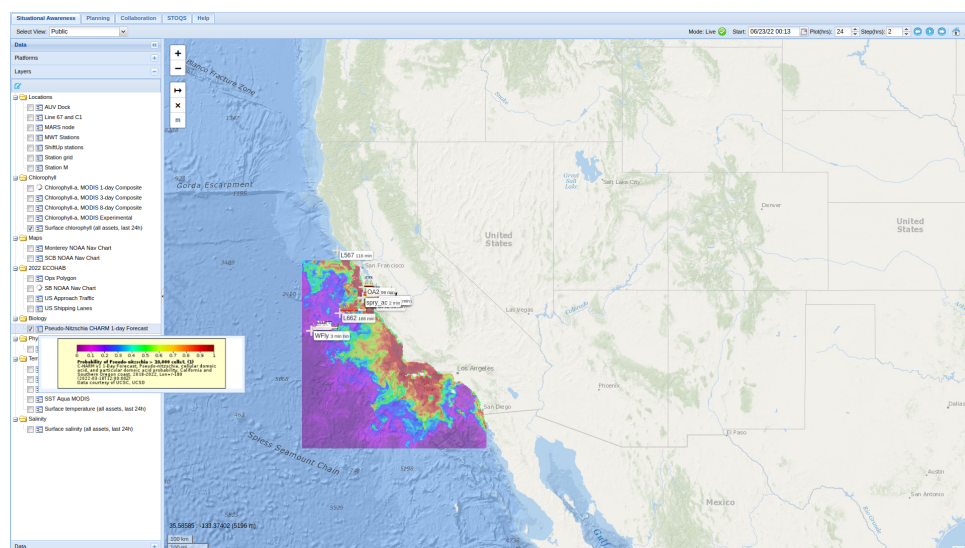


Figure 2.3: ODSS

2.2 UI/UX

2.2.1 Design practices

The subject of design practices and UI design patterns is vast and has many possible applications. For this reason, the main focus of the research was made on human-computer interaction, human-centered design, and decision support and aid.

However, there are design principles that are universal and must be considered when designing an interface. In the book "Designing with the mind in mind: simple guide to understanding user interface design rules" the author provides a summary of user-interface design guidelines and points out the following rules as one of the "two best-known user-interface guidelines": [20]

- Strive for consistency;

- Cater to universal usability;
- Offer informative feedback;
- Design task flows to yield closure (tasks need a beginning, middle, and end);
- Prevent errors;
- Permit easy reversal of actions;
- Make users feel they are in control;
- Minimise short-term memory load. [32]

These principles are important to follow; however, Johnson J. describes that it is also essential to consider "the human perceptual and cognitive psychology that underlies interaction design guidelines". [20] For Johnson, design guidelines should consider the operator's perception, learning, reasoning, memory, and how they convert intentions into actions.

2.2.2 Maritime UI

An area of interest for this dissertation is UI being used in the maritime domain. Interfaces being used in this domain range from military interfaces to ports[5] and docks. These interfaces usually require operators to do critical tasks that require well-informed decisions. As such, the design of human-computer interfaces in this domain requires special attention to the operator's situational awareness and workload.

Nordby et al. point out that "there is a gap between state-of-the-art web-based design support and design support specific to the maritime industry" [24], as such, they propose a maritime design system, which will include "design guidelines aligned with maritime regulations, maritime-specific content [...]"[24]. Although these guidelines are specific to multivendor ship's bridge systems, there are parallels that can inform future decisions on iconography, theming, and colour choices.

An example of the importance of the design stage of maritime interfaces is the case of "Unmanned Surface Vehicle Human-Computer Interface for Amphibious Operations" where the authors worked on a multiyear project to improve an operator's performance when using an interface, Multi-Operator Control Unit (MOCU). In order to achieve their goal of having one operator conducting multiple USV operations it was necessary to "[...] evaluate the effectiveness of the current user interface to gain insights into design improvements".[26] The evaluation of MOCU's effectiveness was done via measuring situational awareness and workload values among other performance indicators. These values informed the iterative design process that followed, in which the team identified "key functions where the current interface may be inefficient, complicated, or perhaps increase workload or fail to provide adequate decision support" [26] and developed several design concepts to tackle these issues.

Another example of the impact of design in maritime interfaces is "Exploring Ecological Interface Design for Future ROV Capabilities in Maritime Command and Control" where the authors

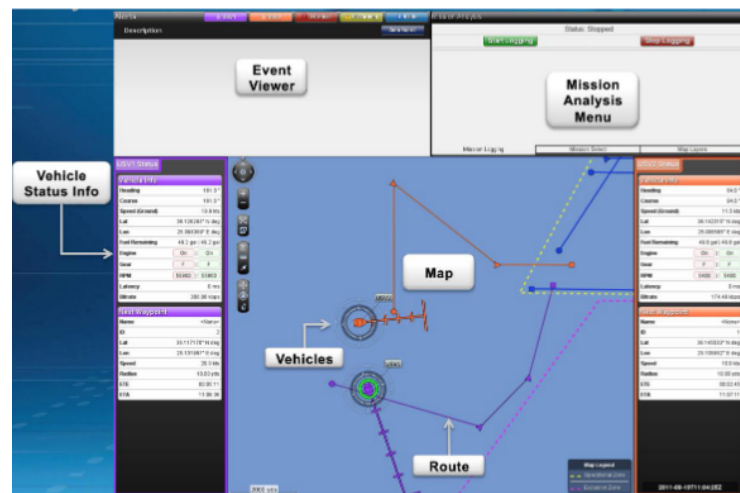


Figure 2.4: MOCU - Vehicle and mission status

suggest applying Ecological Interface Design principles in ROV UI's to reduce operator workload and increase performance. [12]

2.2.3 Ecological Interface Design

Ecological Interface Design (EID) [1, 23, 34, 3] originates from the ecological school of psychology that studies human-environment relationships.

The main goal of Ecological Interface Design is to provide decision-making and problem-solving support in order not to force "cognitive processing to a higher level than tasks require". [12]

The challenge presented by Ecological Interface Design is to ensure that the "constraints contributed by the interface are tailored to the constraints of both the work domain and the cognitive agents".[2]

The principles of ecological interface design are:

- The interface design shall be user-oriented;
- The interface design shall comprehensively analyse the working domain of the system;
- The interface shall provide enough feedback and appropriate information;
- The mapping from information to interface shall be consistent and unique;
- The interface design shall adopt different levels of abstraction to provide information and support people's decision-making activities. [36]

2.3 Situational Awareness

There is no universally accepted definition of situational awareness. However, M. Endsley defines situational awareness as “[...] the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future”.[8]

Measuring situational awareness is a valuable tool to evaluate user interfaces and their impact on the user’s experience. As such, several techniques have been implemented to assess a person’s level of knowledge and understanding of their current situation.

2.3.1 SAGAT

Situation Awareness Global Assessment Technique (SAGAT) is a freeze probe technique that allows the measurement of a subject’s situational awareness in a simulated situation. The freezing of the system is done at random times, during the simulation, and the operators are questioned regarding their perception of the system. "SAGAT is one of the earliest and most widely used measures of SA." [9]

Advantages:

- Useful technique in design evaluation;
- Provides broad testing of SA;
- Removes problems associated with collecting SA data post-trial;
- Provides quantitative results.

Disadvantages:

- Freezing the task disrupts its natural flow;
- Relies on memory, so it can be subject to memory decay;
- Requires the capability to pause the simulation;
- Cannot be applied ‘in-the-field’ or in real-world scenarios.

Moreover, it is possible to apply this technique with concurrent measurements of workload; Endsley states that "Measures of subject performance and workload may be collected concurrently with SAGAT, as no ill effect from the insertion of breaks has been shown." [7] However, the author recommends that some trials should be done without applying SAGAT in order to check this contingency.

2.3.2 SART

Situation Awareness Rating Technique (SART) is a "simplistic post-trial subjective rating technique that was originally developed for the assessment of pilot SA." [28]

SART does not use freeze probe techniques, thus eliminating any breaks while the task is being performed. SART is a post-trial technique where the operator provides their rating on a series of dimensions after having completed a specific task. "The main advantages of SART is that it is easy to use and can be administered in a wide range of task types." [10]

Advantages:

- Easy to administer in a wide range of task types;
- Does not require halting the task;
- Can be used in real-world tasks as well as simulations.

Disadvantages:

- Inability of subjects to rate their own levels of situational awareness;
- Subjects might relate their performance to the values of situational awareness;
- Might influence workload;
- Problems with the collection of data post-task, such as memory degradation and poor recall.

2.4 Workload

B. Cain defines workload as "[...] a mental construct that reflects the mental strain resulting from performing a task under specific environmental and operational conditions. . .".[4]

Much like situational awareness, measuring the workload of performing a specific task can elucidate the impact of using a particular interface on the user's experience. Measuring workload can give information regarding the mental effort necessary for an operator to perform a particular task while interacting with an interface.

2.4.1 NASA-TLX

NASA Task Load Index [17] is a technique that allows the measurement of an operator's workload when performing a task while interacting with various human-machine interfaces. NASA TLX uses a rating scale to measure six workload dimensions to derive an overall workload score based on weighted averages. The six subscales that are part of the questionnaire are mental demand, physical demand, temporal demand, performance, effort, and frustration.

Advantages:

- Widely accepted in the research community;

- Quick and easy method of estimating workload;
- Easy to apply, software available (Figure 2.5);
- Multi-dimensional approach.

Disadvantages:

- Participants may have forgotten relevant workload aspects;
- Workload ratings may be correlated with task performance;
- Sub-scale ratings could be repetitive.

"After nearly 20 years of use, NASA-TLX has achieved a certain venerability; it is being used as a benchmark against which the efficacy of other measures [...] recommended for use in situations as diverse as aircraft certification, operating rooms, nuclear power plant control rooms, simulated combat, and website design." [16]

The screenshot shows a mobile application interface titled "Rate task experience". It contains the following sections:

- NAME:** A text input field with the placeholder "Please enter your name".
- TASK:** A section titled "Read the user guide" with instructions: "Select 'User Guide' in menu and read instructions and tips for using the application."
- DATE:** A field showing the date and time "2018-09-12 20:56".
- Mental Demand:** A section titled "How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc)? Was the task easy or demanding, simple or complex, exacting or forgiving?". It features a horizontal slider scale from "Very Low" to "Very High" with a red marker positioned approximately at the 75% mark.
- Physical Demand:** A section titled "How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc)? Was the task easy or demanding, slow or brisk, calm or strenuous, restful or laborious?". It features a horizontal slider scale from "Very Low" to "Very High" with a red marker positioned approximately at the 25% mark.

Figure 2.5: NASA TLX - App

2.4.2 SWAT

Subjective Workload Assessment Technique (SWAT) [29] is another tool to measure workload in which subjects are asked to rate the workload of a task based on the dimensions of time load, mental effort load, and psychological stress load. The administration of the SWAT technique is a two-phase procedure, the first being scale development, and the second event scoring.

Advantages:

- Generates a simple description of workload;
- Non-intrusive method of evaluating workload.

Disadvantages:

- Time consuming card sorting procedure;
- Low sensitivity for low mental workloads.

2.5 Performance Metrics Summary

In summary, in order to measure how a system's design affects the user's experience, it is necessary to have performance metrics that reflect the domain that is being studied. In this case, those indicators are the values of situational awareness and workload of an operator performing a task.

For measuring situational awareness, we chose the SART technique due to its ease of use, high versatility, and, most notably, the fact that it does not require halting the task. As for workload, we chose to apply the NASA TLX technique because of its validity and applicability across multiple domains.

Chapter 3

Initial Approach

This chapter describes the initial approach to the problem. It will focus on what was done to analyse the working domain of the system and gather users' feedback when interacting with Ripples. There is also a description of the familiarisation process with the interface and source code and mock-ups for an initial interface iteration.

Firstly, we developed a questionnaire addressed to the users of Ripples within the LSTS. This questionnaire aimed to understand the user and their experience in maritime campaigns as well as their requirements and feedback when using Ripples in missions.

Secondly, we analysed this service's functionalities, features, class diagram, and database schema.

Lastly, we conducted an exploration of Ripples' interface with the intent of better understanding the requirements and functionalities and finding any unidentified problems. Having gathered all this information, we did some mock-ups that would tackle some of the design flaws.

3.1 Analysis of the requirements - User Questionnaire

This questionnaire aimed to gather information from the users of Ripples to aid in the design process. With this questionnaire, we intended to collect the end user's views on requirements that may improve operation flow and the tool's scalability.

Four people answered the questionnaire and identified their role in a campaign as an operator. They had all used autonomous and/or unmanned vehicles or systems when performing maritime operations. They all had experience of using these vehicles in a team with software support in the form of a dashboard tool or interface.

The following are some of the questions and answers that helped us better understand the user's requirements and overall experience when using Ripples in an operation campaign.

Question: Please list (up to a maximum of 6) mission moments/situations where having the ability to supervise and/or monitor all said assets in a centralized manner would be beneficial.

Answer(s):

- When third parties... either vessels or unmanned systems interact/join the operational scenario; When the assets are performing a joint collaborative task.
- Collisions and heavy maritime traffic.
- If a large area is to be covered, having this ability is quite important cost-effective wise. Also in complex environments that require multiple assets it's vital to have a synoptic view / control of the complete theater of operations to guarantee the safety of all assets and help "de-conflict" potential issues (for instance, if one asset has a problem you can easily replan or adjust other assets to adapt to that situation).

Question: Please list (up to a maximum of 6) mission plans/decisions that can be impacted by being able to access/analyze these pieces of data/information in a small enough timeframe .

Answer(s):

- Reduce the length of the plan; send the asset to a secure location.
- Safe plans/areas.
- Avoid collisions.
- In case of oceanography it may impact if the goal of the mission is accomplished in time or not.

Question: Removing bureaucratic or funding issues, list (up to a maximum of 6) biggest difficulties currently felt when doing a maritime campaign, starting off from planning and execution to data/result analysis?

Answer(s):

- Coordination of several teams (joint demonstrations); have situational awareness of atmospheric and underwater conditions.

Question: Which of these difficulties, if any, are related to the software tools used in the campaign life cycle?

Answer(s):

- A lot of this can be solved with proper planing with the "right" people for the application. In terms of software, tools such as annotations on the map, plotting options or layers to plot important information to the mission on the map may be quite helpful in this. But of course the mission requirements need to be studied before hand, so the software team can go through them and analyze if developments are required for the mission, in order to make sure the data collected will be useful.

Question: Could you list some of the functionalities of Ripples that you use in the context of your maritime operations?

Answer(s):

- Locations, weather and AIS layers.
- Check payload data and position of vehicles.
- Iridium/SPOT communications over ripples server. It's specially useful to display the synoptic view of the operations with AIS traffic to partners involved in an exercise, since it can run in a browser in any platform.
- Information layers; collision prediction/avoidance.

Question: Focusing on Ripples, could you list some difficulties when using this software that are impactful in terms of their effects on the scalability of operations [that affects the capability of the operation team to “do more”]?

Answer(s):

- Number of icons/assets.
- Distinguishing what is happening at the moment and from what has happened already.

Question: Could you list some difficulties when using this software that are impactful in terms of their effects on the smoothness of operations [that makes the operation experience more cumbersome when performing any task]?

Answer(s):

- Number of icons/assets.
- If a lot of layers are selected it becomes quite heavy to run in some machines.

Question: Looking at the points above, from your point of view, what would you consider the biggest User Interface-related feature (functional or non-functional) that could help solve or mitigate this reality?

Answer(s):

- Group, hide or redraw some assets.
- Make it lighter in terms of resources consumption (if possible) and clean-up the sometimes "excessive visual clutter". Make what is coming through in "real-time" more evident and discard (gray out or hidden layers by option perhaps?) what has already happened so it's easy for the user to perceive the relevant information.

3.2 Analysis of the system

Ripples is a web app that uses the react¹ library for the frontend, it also uses leaflet and react-leaflet² to display the maps and map layers. It uses redux to store states and perform actions that modify the state tree. Regarding the backend, it is being run on springboot³.

¹<https://reactjs.org/>

²<https://react-leaflet.js.org/>

³<https://spring.io/projects/spring-boot>

The system analysis covered the database’s relational schema, class diagrams, and familiarisation with the source code. This process intended to aid with understanding how Ripples worked, what information was available and its relations.

Ripples is a complex system that gathers information from various different sources. Some of the most relevant elements to Ripples were identified during this analysis; these elements were the **AIS Ships**, the **Assets** (Vehicles), the **plans** and their relation to the vehicles and lastly the **GeoLayers** (information layers and maps).

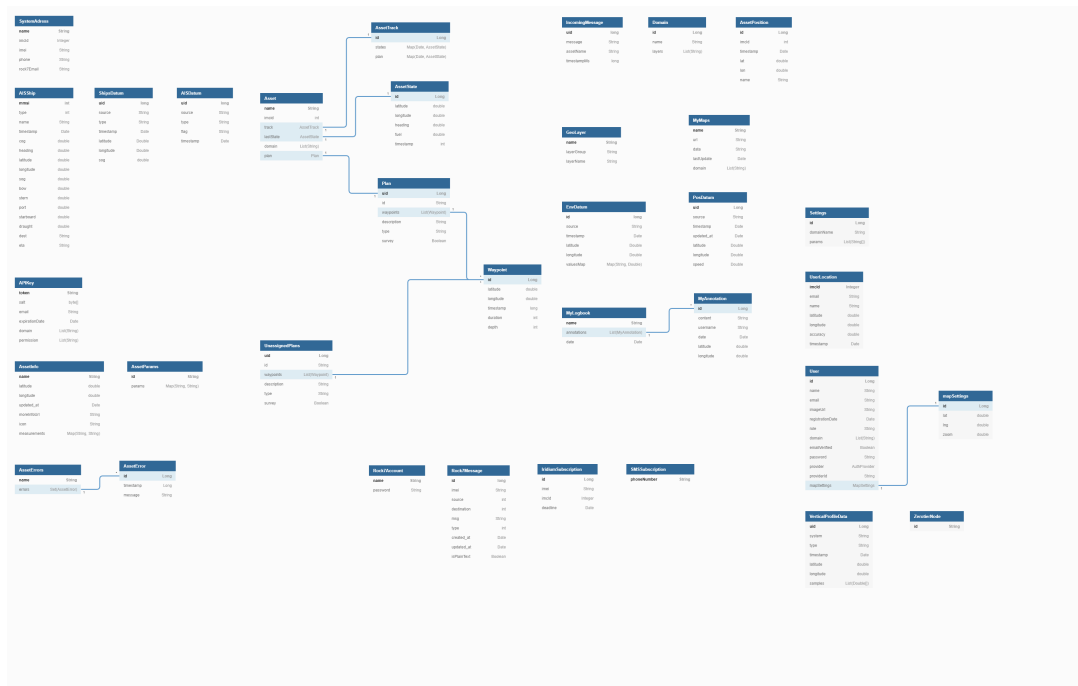


Figure 3.1: Database Schema

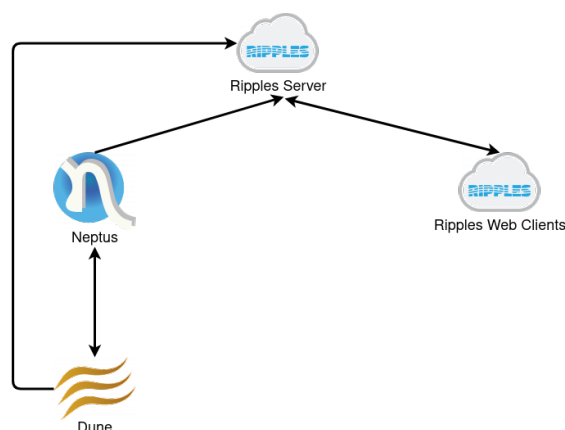


Figure 3.2: Communication Diagram

3.3 Exploring the interface and initial mock-ups

In order to better understand other unidentified problems, it was necessary to explore the interface and its functionalities. This brief exploration phase led to finding a few inconsistencies and problems with the interface, such as when having multiple active layers, it was hard to interpret them. This stage also led to the discovery that only one layer legend was displayed when multiple were selected. These problems led to a few mock-ups, which were done to solve these issues.

The original idea to mitigate the layer visibility problems was to create a new side element that would provide the possibility to define the active layer's priority as well as set their opacity (Figure 3.3). This new element would display the active layers, which would be draggable, and their order would be reflected in the map. Another functionality of this element would be the possibility to define individual layer opacity, which would leave to the users to dictate their preferences.

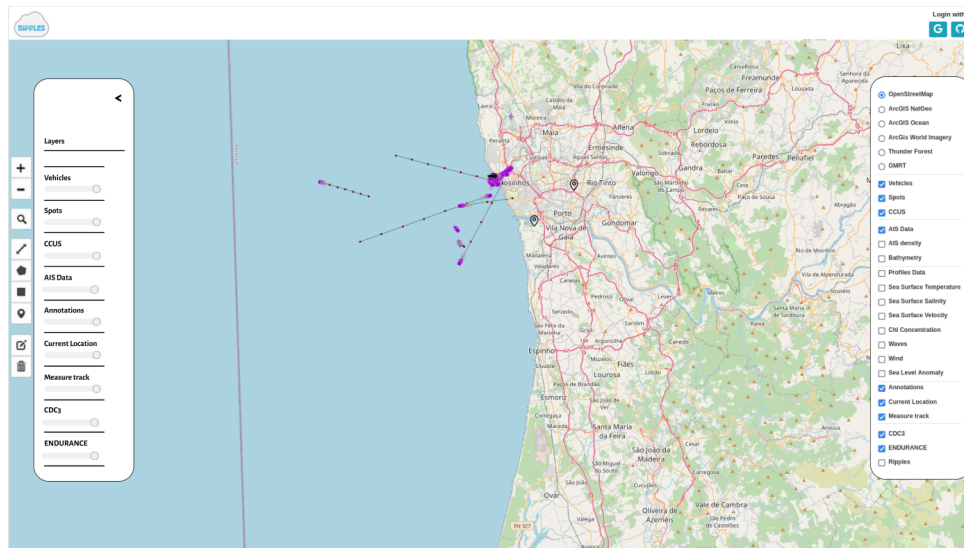


Figure 3.3: Ripples mock-up for layer priority

Regarding the legends and their stacking, an initial solution was to migrate them to the bottom of the web page, where it would be possible to display multiple legends without them overlapping. The problem with this solution is that this element would reduce some of the map's visibility (Figure 3.4).

Some layers, when selected, would also not display any information, such as Wind and Chl Concentration, or would display incorrect information, such as Sea Level Anomaly (Figure 3.5).

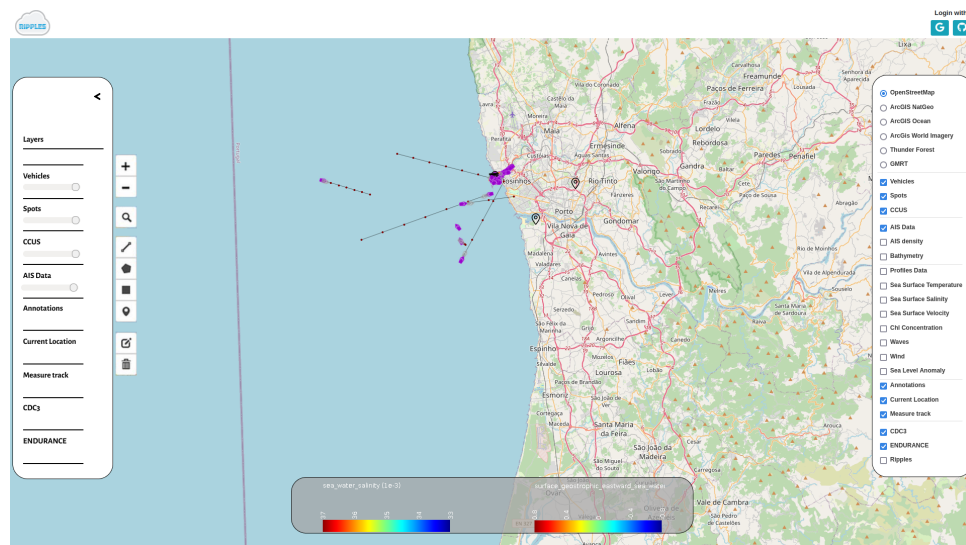


Figure 3.4: Ripples mock-up for layer legends

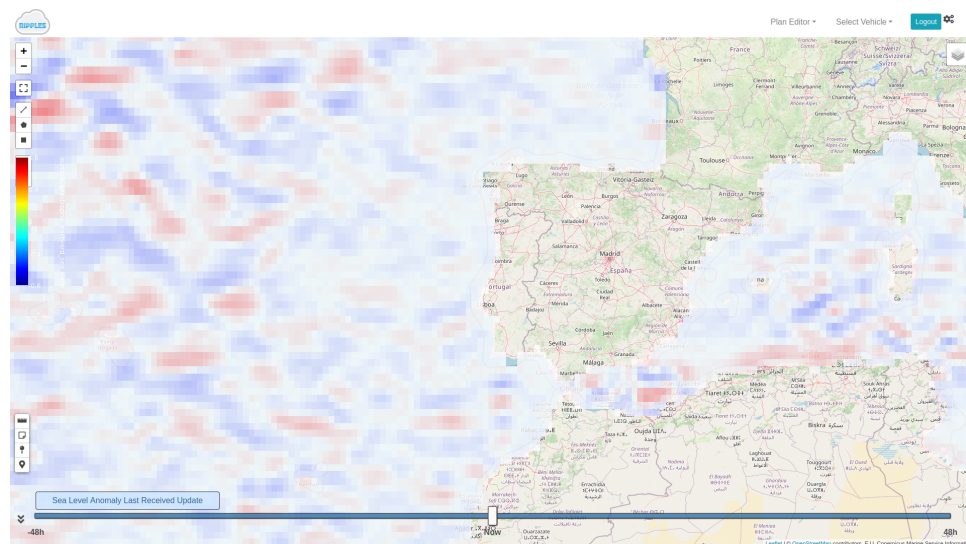


Figure 3.5: Ripples - Sea Level Anomaly

Chapter 4

Implementation

This chapter includes a detailed description of the work done in order to obtain the finalised version of Ripples.

The implementation phase was guided by the principles of ecological interface design and some of the broader design principles previously identified in chapter 3. The questionnaire was an important tool to help us focus the first design iterations on the problems identified by the users.

4.1 Ecological Interface Design

This first section will focus on some of the ecological design interface principles that were applied to Ripples' interface; they were both helpful in finding bad design practices within the interface as well as correcting these problems. The following three rules stood out within the context of Ripples and the laboratory's requirements for this software.

The interface shall provide enough feedback and appropriate information;

The mapping from information to interface shall be consistent and unique;

The interface design shall adopt different levels of abstraction to provide information and support people's decision-making activities.

The interface shall provide enough feedback and appropriate information

4.1.1 AIS Ship highlight

An AIS Ship is a vessel that is equipped with an Automatic Identification System [21]. This system contains a transceiver that transmits the ship's position, course, and speed, among other relevant information. This information is collected by Ripples and displayed in a layer (AIS Data). This information is relevant in campaign settings in order to maintain situational awareness of possible obstacles that might cross the vehicle's plan (path), which would endanger it. When clicking on the icon of an AIS Ship its information is displayed; however, there is no visual feedback of which ship is selected (Figure 4.1).

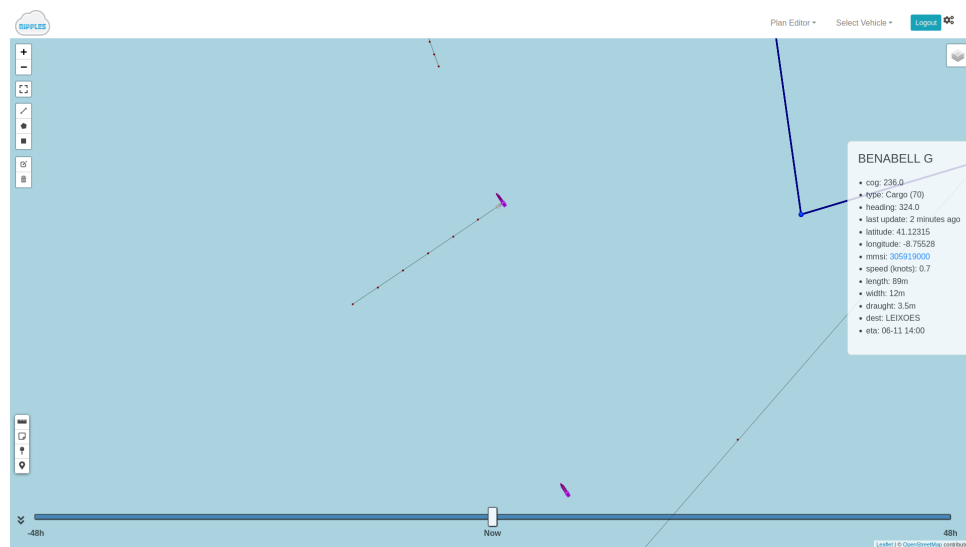


Figure 4.1: AIS Ship - Old version

Moreover, the information box would be overlapped by the sidebar if it was open. The current design highlights the current ship by changing its colour to orange. The information box is now also reactive to the state of the sidebar (Figure 4.2).

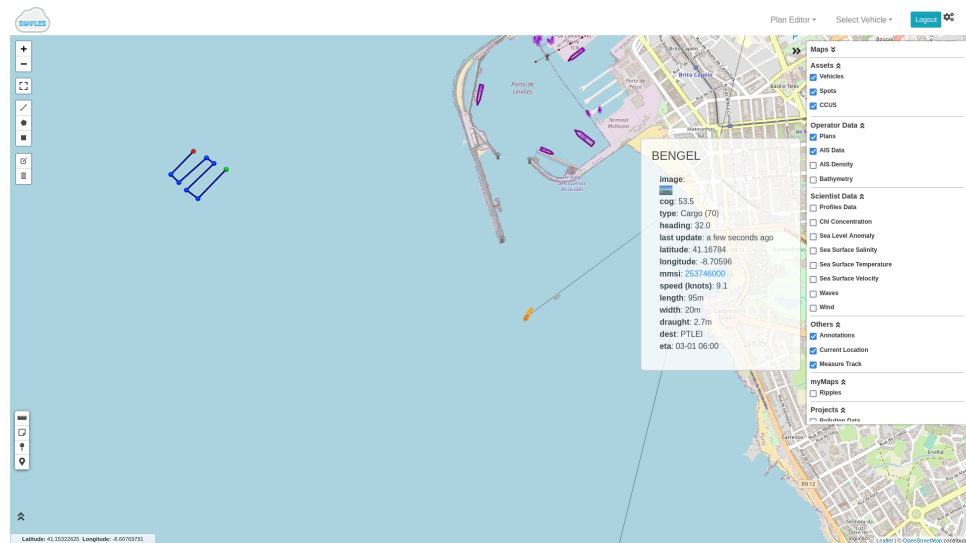


Figure 4.2: AIS Ship - New version

4.1.2 Past and future prediction feedback

Ripples has a slider at the bottom of the page, allowing the user to view past and future predictions of the positions of AIS Ships. When displaying an AIS Ship, two icons are displayed, one for its current position (purple icon) and, in the old interface, a grey icon for an estimated position.

Moving the slider will show the estimated position according to the time that the slider is set to; this means that the grey icon's position will reflect the value set on the slider (Figure 4.3). It

was hard to track the movement of the grey icon due to its opacity and low contrast. Thus it made it hard to follow the icon's movement. This issue worsened because the icon representing the current position was very contrasting in comparison to the grey icon, making it harder to follow the movement.

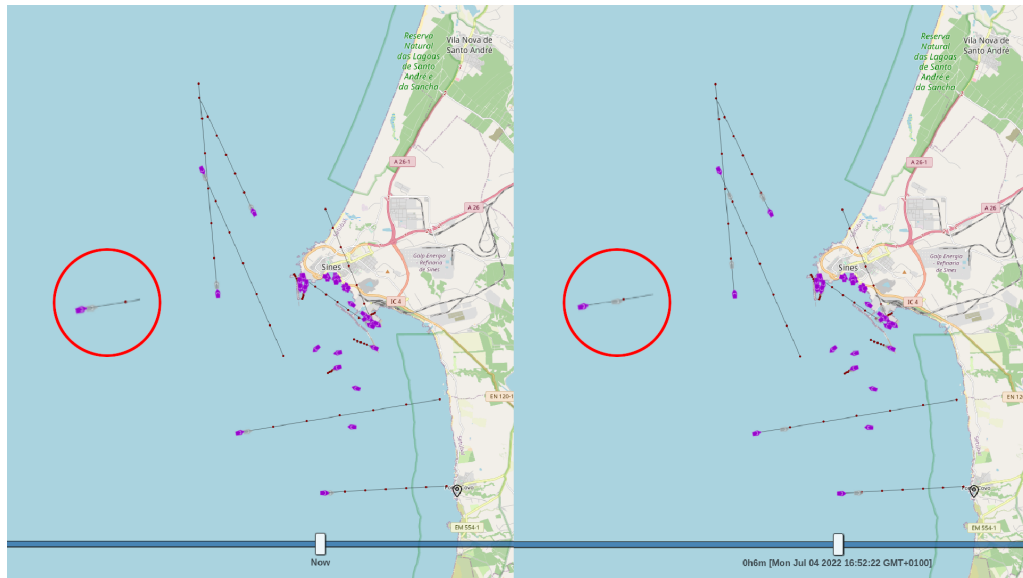


Figure 4.3: AIS Ship Prediction - Old version

The current solution reduces the opacity of the current position icon so that it is not the focus, as the intention is to visualise the predicted position. Furthermore, the icon reflecting the predicted position turns purple and increases opacity (Figure 4.4). The intended behaviour is to show better visual feedback to the user that the slider is affecting the ship's predicted position and that their state is changing.

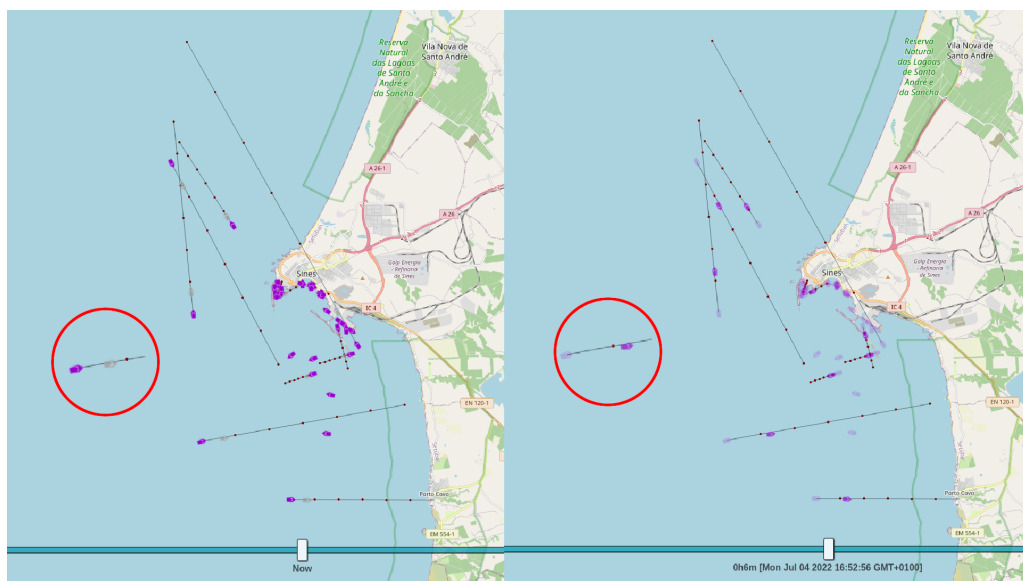


Figure 4.4: AIS Ship Prediction - New version

4.1.3 Winding number algorithm

When showing AIS Ships, there are also lines being drawn that show the predicted trajectory, based on the ship's AIS data, for the next hour. This trajectory is divided into intervals of 10 minutes; for each interval, it is verified if this point is within the boundaries of mainland Portugal. If a certain predicted point was within mainland Portugal, then the trajectory line would be drawn from the current position of the boat until the last predicted point that did not cross land.

The boundaries are represented as a polygon, and the ray casting algorithm was used to check if a point was contained in the polygon. The algorithm, however, has the limitation of only working with simple polygons (polygon that does not intersect itself and has no holes). Unfortunately, the polygon used to represent Portugal's outline is a non-simple polygon, meaning that the ray-tracing algorithm in use could not correctly compute if these points were contained within this polygon. Because of that, some boats were displaying their trajectory on land, contrary to the intended behaviour (Figure 4.5).

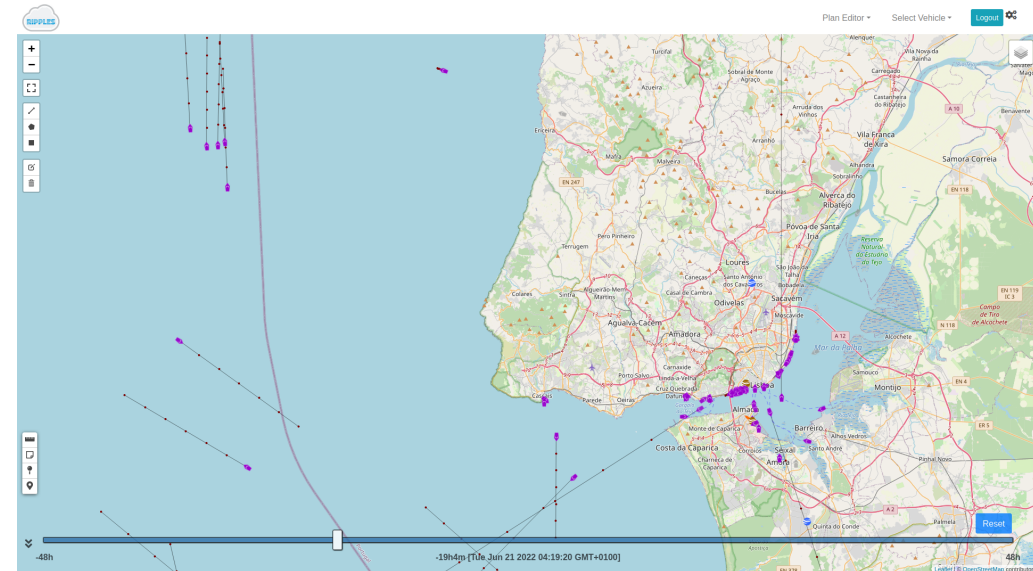


Figure 4.5: AIS Ship - Trajectory and past positions

The winding number algorithm does not have this limitation, meaning that it is better suited for this case. It is, however, computationally more intensive than the ray casting algorithm. This algorithm calculates the winding number of a point R with respect to a polygon P . The winding number "is the number of revolutions made around that point while travelling once around P " [18]. If this number is zero, the point will be outside the polygon. Dan Sunday implemented a lighter version of this algorithm in his book *Practical Geometry Algorithms with c++ code* [33]; this implementation (Listing 4.1) does not require heavy computation, so it was the one that was used to replace the previous algorithm.

```

1  // check if a point (p2) is left of two points (p0,p1)
2  private isLeft(p0: any, p1: any, p2: any) {
3      return (p1[0] - p0[0]) * (p2[1] - p0[1]) - (p2[0] - p0[0]) * (p1[1] - p0
        [1])
4  }
5
6  private pointInPolyWindingNumber(point: any, polygon: any) {
7      if (polygon.length === 0) {
8          return false
9      }
10
11     const newPoints = polygon.slice(0)
12     newPoints.push(polygon[0])
13     // winding number counter
14     let wn = 0
15
16     // loop through all edges of the polygon
17     for (let i = 0; i < polygon.length; i++) {
18         if (newPoints[i][1] <= point[1]) {
19             if (newPoints[i + 1][1] > point[1]) {
20                 if (this.isLeft(newPoints[i], newPoints[i + 1], point) > 0) {
21                     wn++
22                 }
23             }
24         } else {
25             if (newPoints[i + 1][1] <= point[1]) {
26                 if (this.isLeft(newPoints[i], newPoints[i + 1], point) < 0) {
27                     wn--
28                 }
29             }
30         }
31     }
32     // the point is outside only when this winding number wn===0, otherwise it'
        s inside
33     return wn !== 0
34 }

```

Listing 4.1: Winding number algorithm

There are still edge cases where the points that will be checked when drawing the AIS Ship's predicted trajectory are in the ocean, but their resulting line crosses land mass. Moreover, the polygon's definition is limited. As such, some points will be inland but are not covered by the polygon, as is observable in Figure 4.7, where the line does cross land.

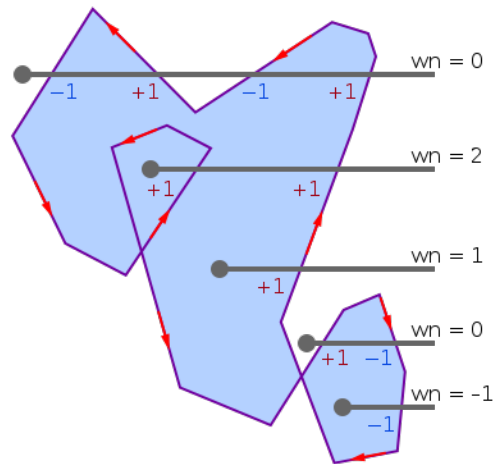


Figure 4.6: Dan Sunday's winding number algorithm

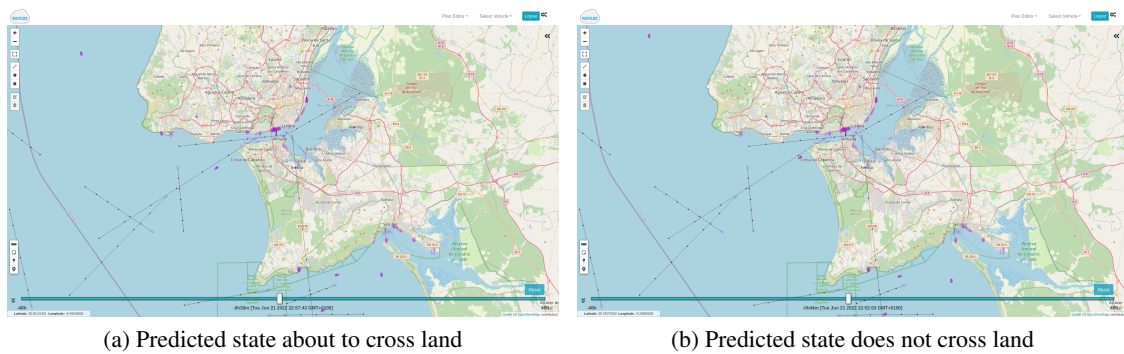


Figure 4.7: AIS Ship - Lines and predicted states (New algorithm)

The mapping from information to interface shall be consistent and unique

4.1.4 Layer legend consistency

When a layer was selected in the old interface, a legend would appear in the highlighted area of the interface (Figure 4.8). When multiple layers were selected, it would only show the legend of the latest selected layer, and if that layer was toggled off and there were other selected layers, no legend would be shown. This behaviour made it impossible to read more than one layer simultaneously. The original solution for this problem was to create a specific area for the legends where there would be space to show multiple legends (Figure 3.4).

The legend was an image that was retrieved via the Copernicus API with the GetLegend-Graphic request. In this request, it is possible to select the legend style (legend colour range) and its scale. Most layers use a rainbow style in their scale colour range (Figure 4.9). This was the default style value for the request, and when an invalid style was provided, it would default to this style. This behaviour was observable when selecting the Sea Level Anomaly layer, where the

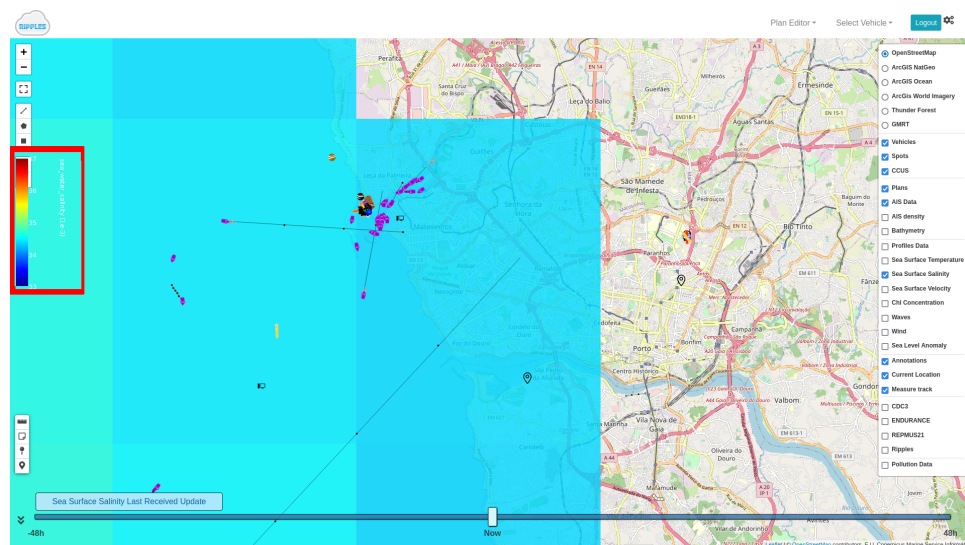


Figure 4.8: Ripples - Legend area

GetLegendGraphic request did not correctly retrieve the red-blue style.

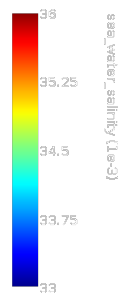


Figure 4.9: Ripples - Legend example

Moreover, the layer legends would be hard to read as the units, and scale values on the image were relatively small and not very contrasting with the background. This meant that in some situations, the scale would display wrong information or would be hard to read.

The solution to this problem was implemented as part of a more significant element, redesigning the layer's sidebar. The layer sidebar redesign will be explained later in Section 4.2, however here, we will explain what is relevant to this problem. The layer's legend was moved to the sidebar, under the respective layer, and is only shown when the layer is active. This way, multiple layer legends can be shown simultaneously without occupying more space and adding visual entropy within the map area. Another benefit of this move is that the legend has a solid background, meaning it is easier to read the units and scale values.

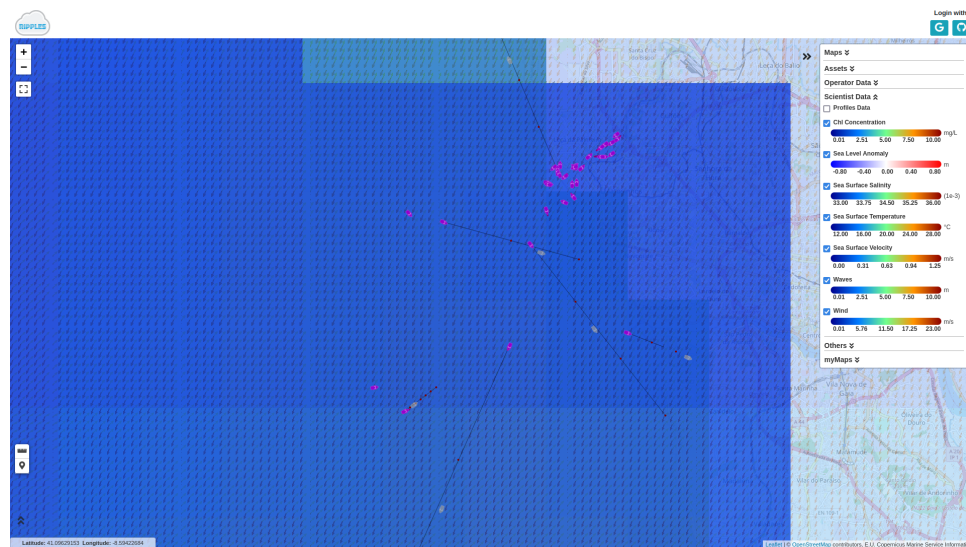


Figure 4.10: Ripples - New legends

The interface design shall adopt different levels of abstraction to provide information and support people’s decision-making activities

4.1.5 Role-specific view

Some of the information displayed was relevant for an operator but irrelevant for scientists and vice-versa. Displaying equally all information caused visual entropy between the different sets of users. One of the goals when redesigning Ripples was to reduce the visual entropy between users. This meant it was necessary to understand the types of users and what information was necessary for each of them. We identified four different users:

- Casual - any user that isn’t logged in or hasn’t been attributed any specific role;
- Operator - users that partake in deploying and controlling the assets;
- Scientist - users that mostly use Ripples as a tool to consult information (can also serve as an operator);
- Admin - people actively developing Ripples or needing special permissions;

Among these users, the ones that represented the most significant differentiation in their requirements were the operators and scientists. For scientists, it is more important to be able to visualise and interpret data from weather forecasting layers or other relevant information regarding the ocean. Whereas, with operators, it is vital to visualise other vehicles that are in the ocean and other uncontrollable variables that affect an asset and might endanger it. As such, it was identified the need to provide different views to the different users, with a particular focus on the operator and scientist roles.

Regarding scientists, it was identified that these users also benefit from the AIS Ship layer, the layer showing ships equipped with AIS trackers. Nevertheless, there is a lot of information being

shown in this layer (Figure 4.11), and it is possible to simplify and reduce the emphasis of this layer to the benefit of these users. By removing the lines that show the ship's predicted trajectory whilst maintaining any functionalities tied to it and reducing the opacity of the AIS Ship icons, it was possible to reduce the visual clutter associated with the layer (Figure 4.12).

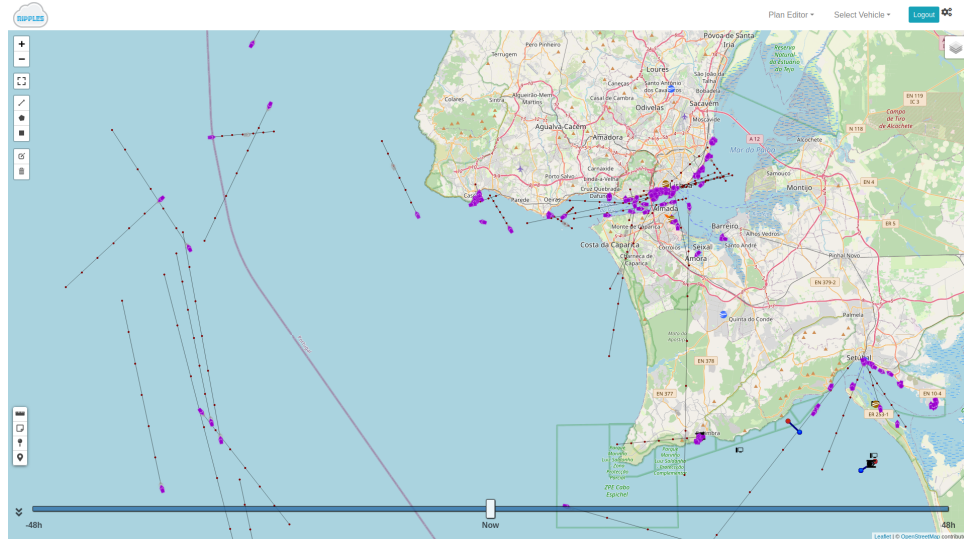


Figure 4.11: Ripples AIS - Scientists old view

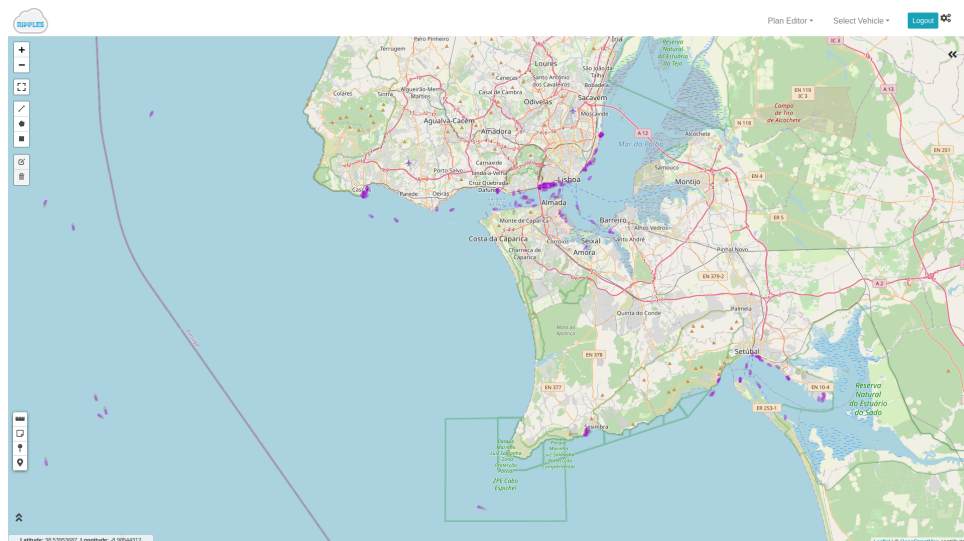


Figure 4.12: Ripples AIS - Scientists new view

For operators, it was identified that the most relevant layers to scientists are not as relevant to operators. Again, this does not mean that these users will not use these layers, and having the possibility to consult them is necessary. A layer such as Sea Surface Velocity contains relevant information (Figure 4.13) to an operator when deploying a LAUV (Light Autonomous Underwater Vehicle) as the current of the ocean will affect the vehicle's position. Notwithstanding, we lowered the opacity of these layers, in some cases, we halved the opacity compared to a scientist's view. Each layer's opacity value was explored individually to be able to interpret the displayed information whilst reducing the visual clutter (Figure 4.14). This behaviour had the intended function of keeping the operator's focus on what is surrounding them and the vehicles that they have deployed while maintaining access to other less-used functionalities.

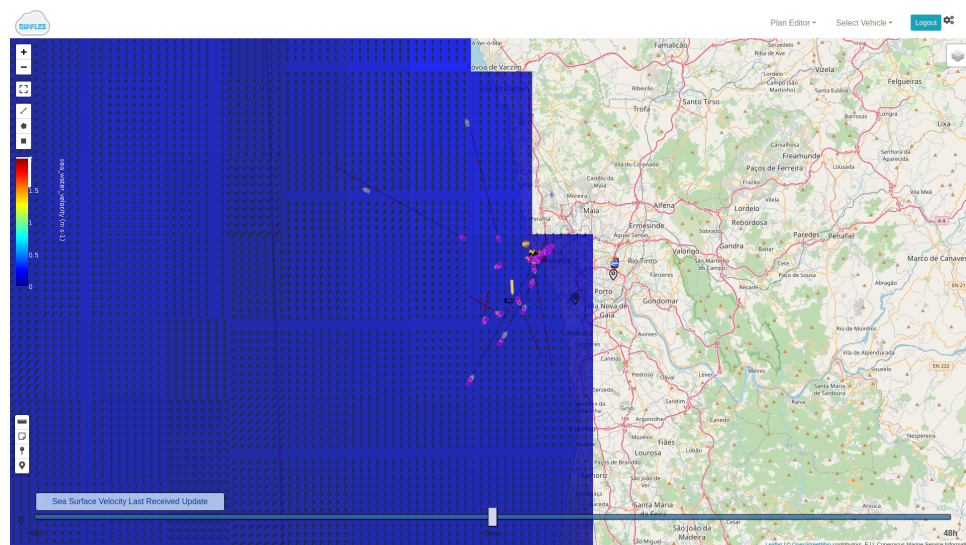


Figure 4.13: Ripples - Operator old view

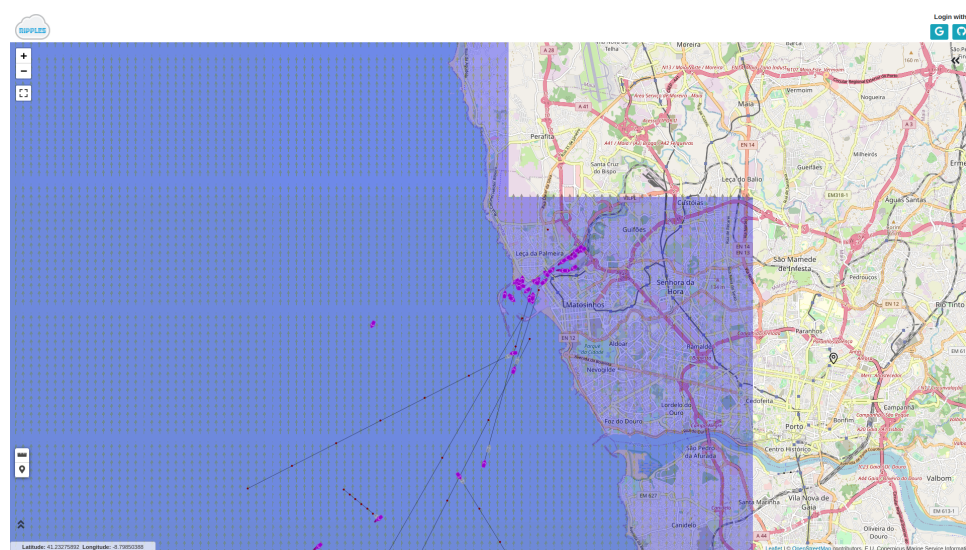


Figure 4.14: Ripples - Operator new view

Lastly, it was identified the necessity to provide some flexibility to the layer selection sidebar. This element contains a variety of different layers with different purposes and relevance to different users. As such, some changes have been made to reduce entropy between users. The layer selection sidebar is now divided into layer groups; each group is identified and can be closed or expanded per the user's preference. Different users have different layers on the sidebar, some granted by special permissions and some related to their role. Regarding the role-specific view, the sidebar has different layer groups closed by default for previously identified users:

- Casual - Scientist Data and Operator Data are closed by default;
- Operator - Maps, Scientist Data and "Others" are closed by default;
- Scientist - Maps, Operator Data and "Others" are closed by default;
- Admin - Maps and Scientist Data are closed by default;

Scalability was also a concern when implementing this feature. Consequently, it is easy to add a new role and define what layers will be collapsed by default for those users.

4.2 New sidebar design

The sidebar was one of the elements that was immediately identified as having the possibility of improving. Firstly, the hover functionality (users had to have the mouse pointer hovering over the sidebar for it to be open) that was being used, although practical, was not ideal for the use cases of Ripples. When on a sea campaign, it is common to use Ripples on computers or phones when on a boat, with hard light conditions. Thus, it makes it hard to maintain the dexterity and visual perception to keep the cursor within the bounds of the sidebar. Even in less challenging scenarios, sometimes using this sidebar proved frustrating.

This sidebar was an element from leaflet whose design was the default from this library. The element was also very unorganised; there were small lines separating some layers. These separators were not identified, so the organisation felt arbitrary, and the grouping they intended to create was hard to identify.

The initial idea was to build a new sidebar, creating a new element that was not directly tied to the leaflet/react-leaflet libraries. However, the sidebar is directly tied to the layer creation and rendering processes. So it would be impossible to create an element that was completely abstracted from the leaflet library.

Then we explored the idea of adding elements within the LayersControl element (an element that will render both the layers and the sidebar). This experiment proved unfruitful, as all elements that were not a ControlledLayer, BaseLayer, or Overlay would be removed from the sidebar and rendered outside the viewport. This behaviour meant that there was no way of adding elements to the sidebar because the react-leaflet library would not let the LayersControl element have any child that the library did not specify.

Afterwards, we explored the idea of creating a custom leaflet LayersControl. We explored several methods of implementing a custom leaflet LayersControl, and also found that there were libraries that implemented a custom LayersControl element [30] which would allow for grouping layers, collapsing groups and adding any other HTML or react element without them being ejected from the sidebar (Figure 4.15). At first, implementing a custom LayersControl element seemed like the solution. However, when adding it to Ripples, it did not work. Custom elements for react-leaflet only work with version 3 or higher, all lower versions were too strict to allow for any customisation, and Ripples uses version 2 of react-leaflet.

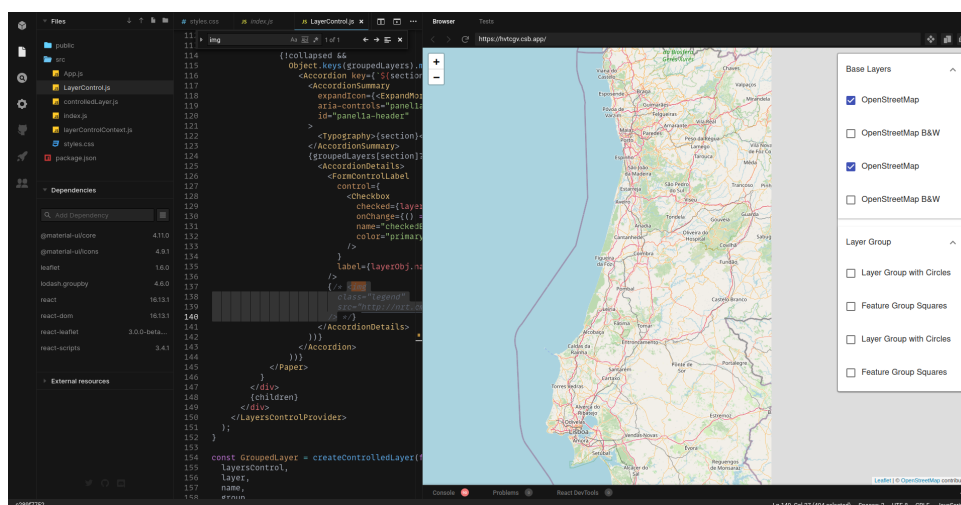


Figure 4.15: Custom LayersControl

Naturally, we tried updating react-leaflet to a stable release of version 3. However, this software uses a lot of third-party libraries, all with different dependencies. So when updating the react-leaflet version, several elements did not work, and there were many compatibility issues. So, we tried to downgrade the version of react-leaflet of this custom element and modify it to work with an older version. As react-leaflet version 2 is too strict and does not allow for any customisation, the custom LayersControl element was not functioning when downgraded to this version.

Finally, we decided to inject HTML elements to the component once it finished rendering; this solution is not as polished as intended but was implemented as a last resource due to the previous attempts failing. Once the LayersControl element has been rendered, it is possible to edit child elements or add new child elements. This meant that we had to use the react-leaflet element and edit it to implement the intended functionalities.

4.2.1 Layer grouping

One of the problems with the LayersControl element was the existing layers grouping. The layers were separated without any identification of what each group represented, and the line separating these groups was hard to notice; thus, the groups were hard to identify and comprehend.

The new sidebar element (Figure 4.16) contains a group label at the top, clearly identifying what each group provides. Next to the label, there is also an icon identifying that the group can either be expanded or closed. The line separating each group has also been made more noticeable, increasing its contrast against the white background. These changes were made to clarify to any user what each layer group is. Another improvement is that new users will be able to navigate the sidebar with more ease.

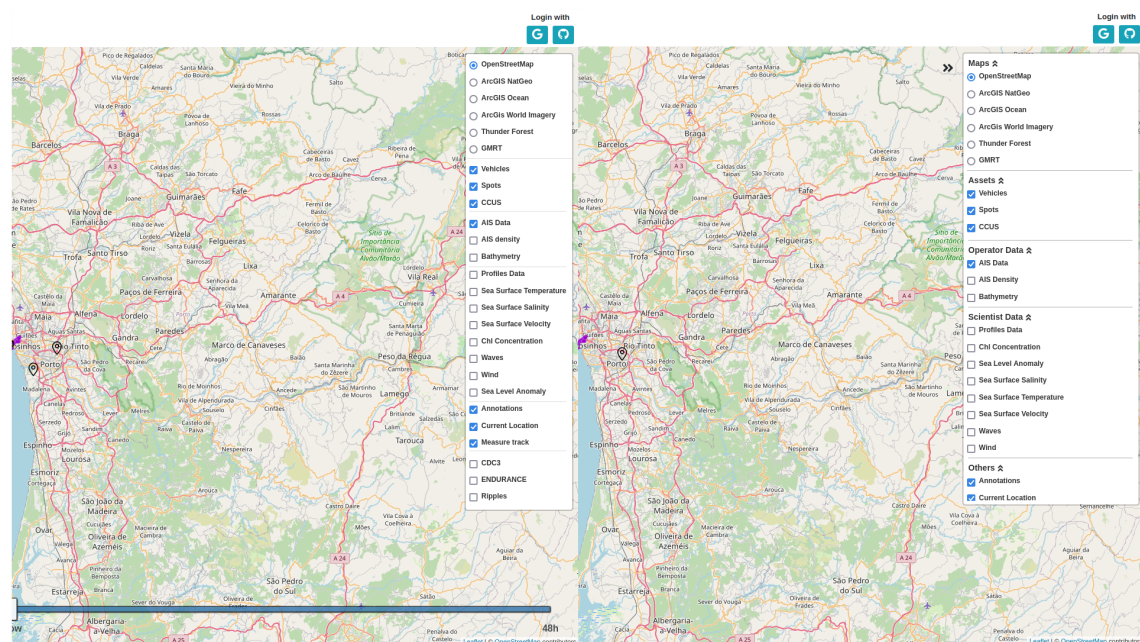


Figure 4.16: Sidebar comparison (Old vs New)

4.2.2 Custom labels

Initially (Section 3.3), the idea for the labels was to keep using the Copernicus API to retrieve the legend of the layer and display them in a different area where it would be possible to show more than one legend at a time (Figure 3.4). However, when exploring this API, some problems were identified, such as style options (the possibility to choose the legend style, size, and orientation) not working. This meant we could not get the image as it was intended, with a horizontal orientation, correct layer colours, and smaller size. For this legend container to work, it would be necessary to resize and rotate the images, which degrades the image's quality when done with CSS. In this case, the image was rendered with a quality that made it impossible or hard to read and interpret.

With the possibility of editing the sidebar, the legend container was abandoned; in favour of showing the legends underneath the corresponding layer selector. The layer legends were done by creating a custom `div`¹ element that would represent the layer's colour range. This component was implemented using CSS and gradient functions. Currently, there are only two different layer styles in use, and both are being handled. This change is done automatically if a layer's style

¹<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/div>

is changed to one of the two current styles. Moreover, the value of the scale and units could be hard to read, depending on what was being shown in the background (Figure 4.17). Moving these elements to the sidebar meant that the background no longer changes, so the legend values are always contrasting, improving their readability (Figure 4.18). An added benefit of this design choice is that the map area is now less cluttered.

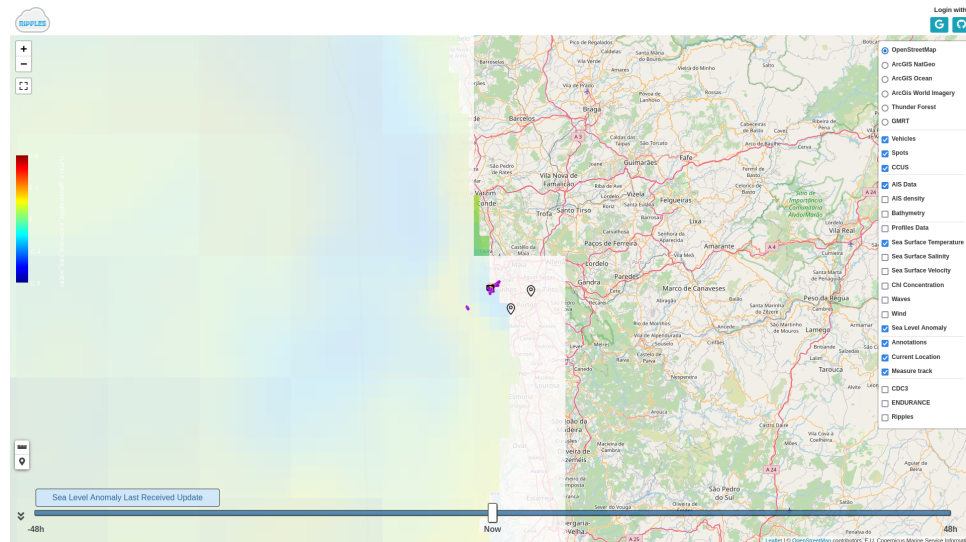


Figure 4.17: Scale values and units - Hard to read

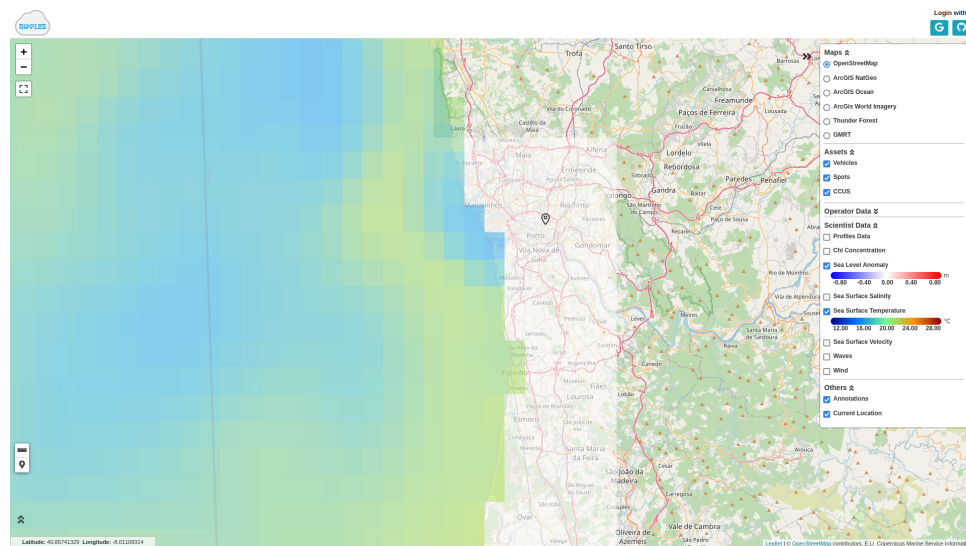


Figure 4.18: Scale values and units - New label design

4.3 Plan feedback

An important feature of Ripples is the possibility to draw campaign plans for multiple vehicles; this feature is helpful in the mission planning stages and during a mission. Furthermore, since

this feature is directly related to the vehicles and their safety, the person drawing the plan must be capable of making informed decisions. As such, we pinpointed two areas where the user could get more feedback. These two functionalities are:

- After creating a plan, being able to provide feedback to the user about the traffic density of the area where the plan is being created;
- For each plan, provide feedback if a AIS Ship is going to cross the plan within a predefined operational time window.

4.3.1 High density areas

When creating a plan, users must identify the dangers of said plan to the vehicle that will be deployed. One of these possible dangers is the area in which the vehicle is being deployed, and if this area is an area of high-density of marine traffic, then the vehicle will be traversing a dangerous area. This traffic density can be visualised via the AIS density layer. However, not all users will consult this layer when drawing a plan. So, we created an automatic warning that is shown after the user creates a plan in a high-density area (Figure 4.19).

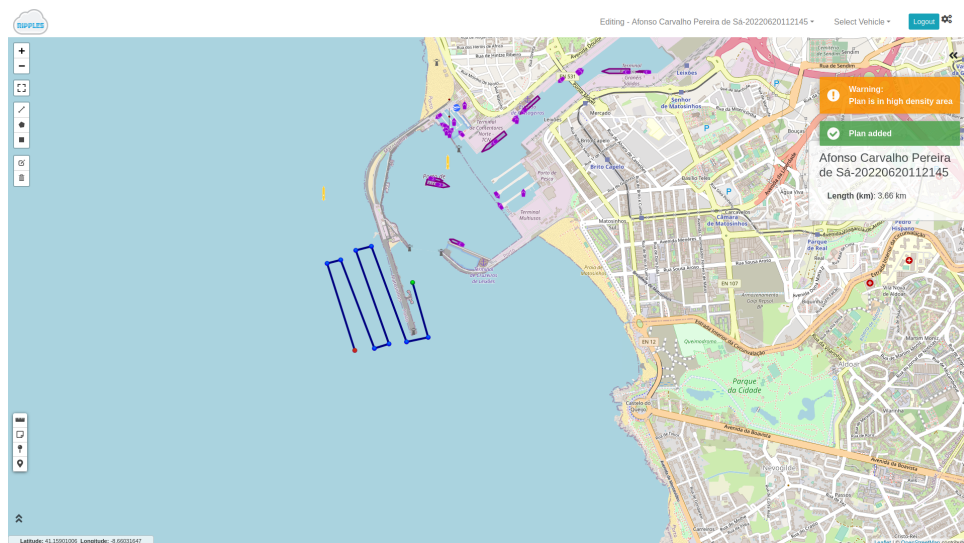


Figure 4.19: Warning message on plan creation

4.3.2 AIS Ship is in operating window

Another possible danger to a vehicle is when an AIS ship is heading towards a plan's trajectory. There is an operating window in which these ships crossing a plan become dangerous to the vehicle associated with that plan. In this case, that operating window was set to ten minutes. This time frame was chosen because it allows the operator enough time to process the information and react accordingly. Using the AIS ship's predicted trajectory, it is possible to verify if the ship's current position and its position in ten minutes crosses any plan's trajectory. When the plan's trajectory is

being crossed by a ship within the operating window, the trajectory will change its colour to red, showing a warning to the user that that plan is in danger of having a ship crossing it (Figure 4.20).

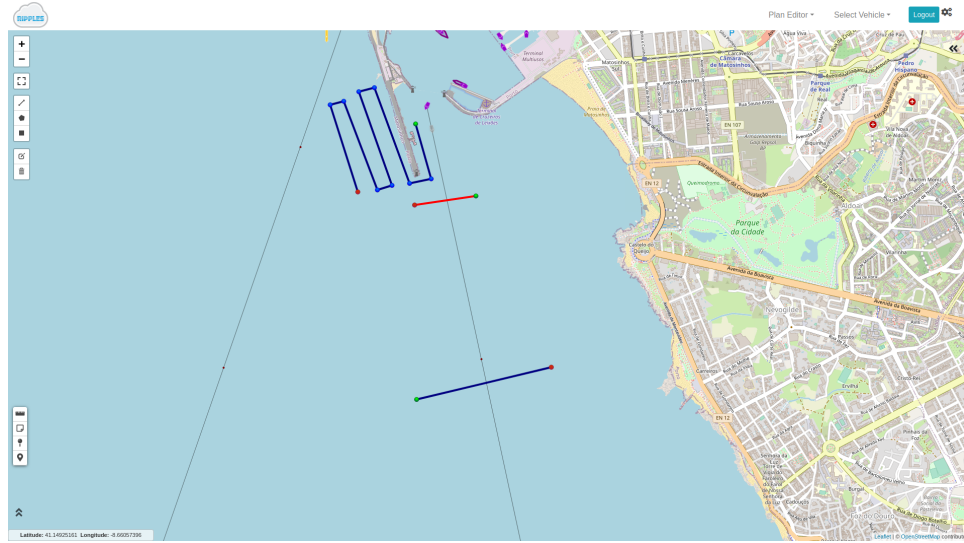


Figure 4.20: Two plans - Warning operating window

4.4 Layer scale definition

Another functionality that was identified was the possibility of displaying some layers with a variable scale. Ideally, this scale would automatically adapt to the position being shown to the user. When exploring the Copernicus API via their web viewer, GODIVA2 (service used as a demo to preview the data provided by the API), we found that they already had this functionality (Figure 4.21). However, this functionality is not provided by their API.

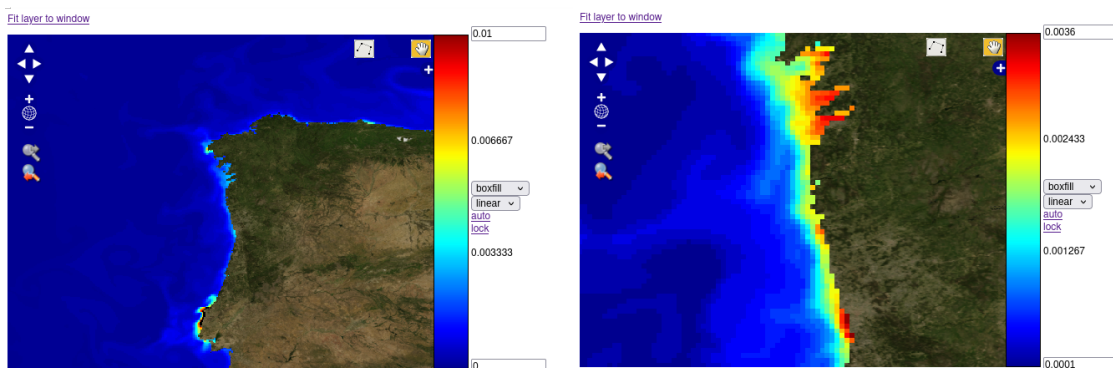


Figure 4.21: Copernicus automatic scaling

Each layer tile (a chunk corresponding to an area in the map) is an image that is retrieved via Copernicus. As an alternative method, we considered averaging the images' red, green and blue values; for layers with a rainbow colour scale, if the blue value was greater than the red or green it meant that the lowest scale value could be raised. Conversely, if the value was greater than the

blue or green channels for the red channel, it meant that the scale's ceiling could be raised. This option had to be ruled out because the react-leaflet library automatically adds the images to the HTML canvas element, making it impossible to retrieve the isolated layer images.

Due to time constraints, we implemented a rudimentary version of the intended behaviour (Listing 4.2). The current functionality is not automatically adjusting the scale according to the values being displayed, but it is adjusting the scale according to the zoom levels and the map position. If the zoom levels are above a certain threshold (greater or equal to six) and the map is centred within a specific range of latitude and longitude (centred around the Iberian peninsula), then the layer's scale can be reduced to show more definition. Currently, this functionality is only being used for the Sea Surface Velocity; nevertheless, it can be expanded to other layers.

```

1 // check if map is centered in a specific area and a certain zoom value
2 if (this.map) {
3   const mapBounds = this.map.leafletElement.getBounds().getCenter()
4   if (34 <= mapBounds.lat && mapBounds.lat <= 48 && -30 <= mapBounds.lng &&
5       mapBounds.lng <= 10) {
6     velocityRange = this.state.currentZoom >= 6 ? '0,1.25' : '0,2'
7   }
8 }

```

Listing 4.2: Layer scale code

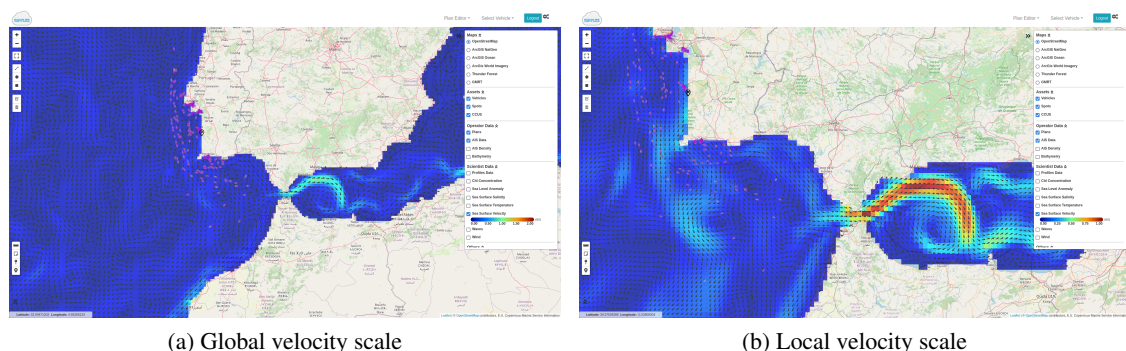


Figure 4.22: Layer scale definition

4.5 Simplified AIS information panel

The AIS Ship information panel was identified as cluttered and hard to read, so we simplified it by removing some superfluous information and changing the order in which the information appeared. Previously the information was scattered with no particular order to it; for example, information about the ship's movement was not grouped (Figure 4.23). We established what order the information should be presented in and grouped it. We also differentiated the information

variables from their values and added, when possible, an image of the ship (Figure 4.24). The rationale behind adding the image was to provide the user with a method to visualise the ship's actual dimensions and make it easier to spot.

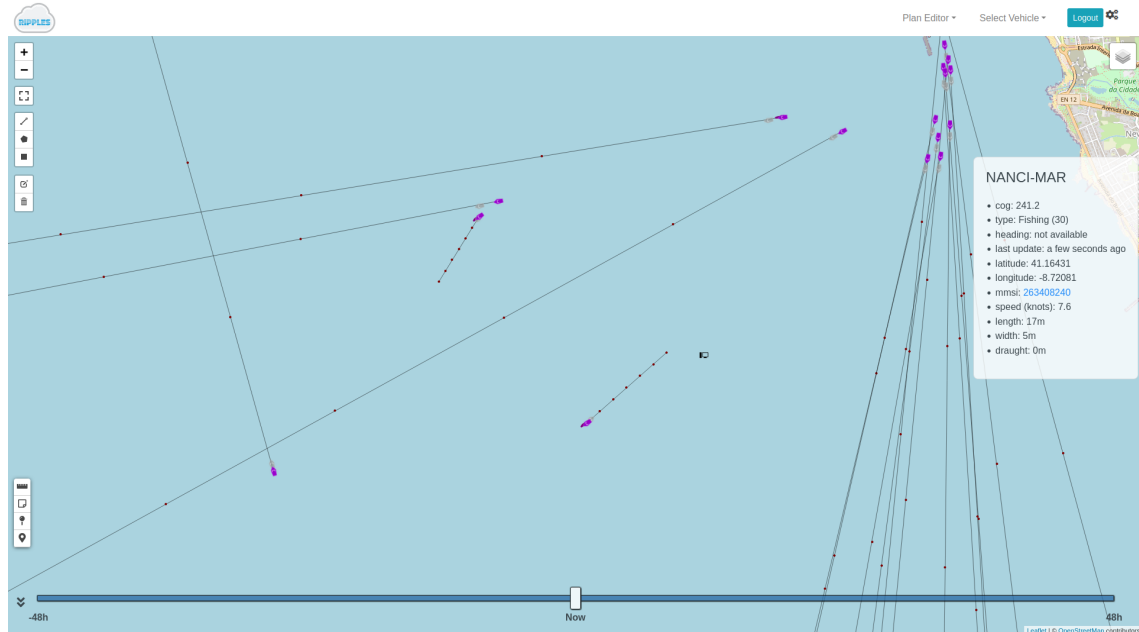


Figure 4.23: AIS information panel - Old

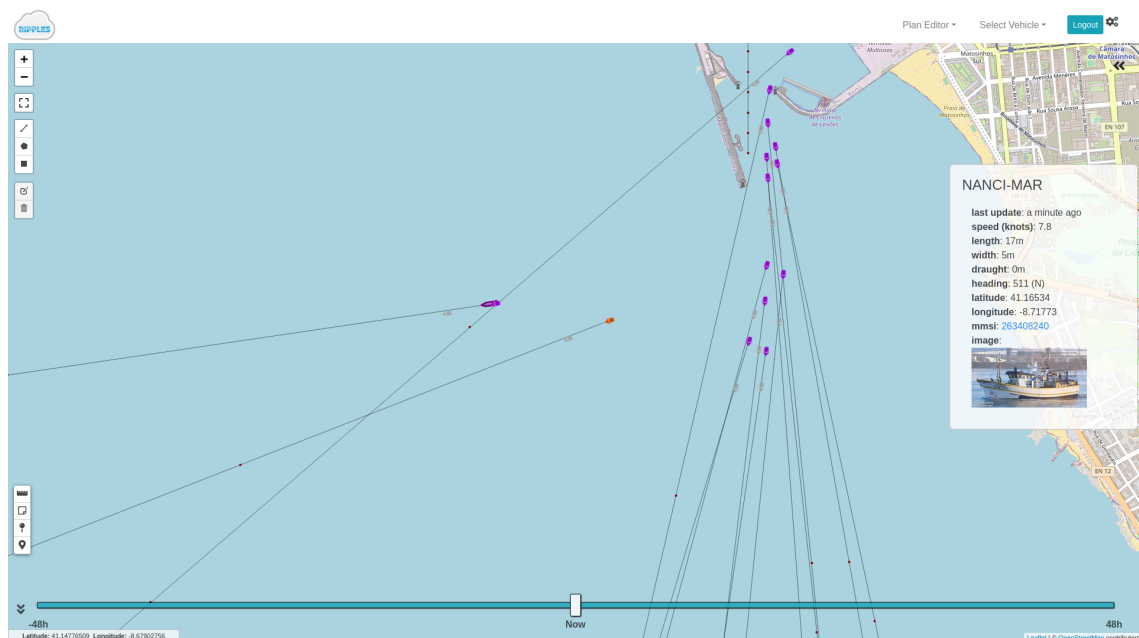


Figure 4.24: AIS information panel - New

Chapter 5

Results and Discussion

This chapter contains an explanation of the performance metrics, information on how the tests were conducted, and an analysis of the results. Upon conclusion of the implementation of the new interface, it was necessary to test both interfaces to compare them and evaluate if the proposed objectives were successfully met. The tests were conducted with personnel from the LSTS, as they required a certain level of familiarisation with Ripples.

5.1 Performance Metrics

The performance metrics for these tests were the **workload values**, obtained via NASA TLX, and the **situational awareness values**, obtained via the SART rating. The NASA TLX rating was chosen to obtain the workload values because of its validity and applicability across multiple domains. The SART rating technique was chosen to measure the values of situation awareness because it did not require halting the tasks, making it compatible with NASA TLX, and because of its ease of application.

5.2 Tests

The tests were conducted on the same setup for all participants, the tasks were the same between versions, and the rating methods were also the same. In order to reduce any bias from performing the same sets of tasks on the old and new interface, we rotated what version the user started with.

In total, the users had to conduct six tasks, one of which had multiple sub-tasks, ranging from operator-oriented tasks to scientist-related tasks. The user's performance was not directly evaluated. However, we did collect the data from failed tasks and other feedback the user gave during the tests.

All tests were conducted at the LSTS, apart from one person that could not test in person. In total, ten users were tested, all from the laboratory, and all had experience and knowledge of the toolchain from the LSTS. The in-person tests were done using two monitors, one for visualising Ripples and the other for showing visual aids to help the user understand the task. The remote test

was done via TeamViewer; the latency throughout the tests was consistent and, as such, can be considered negligible. However, the remote test had an issue where the scroll wheel was not fully working, so the user had to use the controls provided in Ripples to zoom out. Again, this issue was present in both versions so these results were not invalidated.

The tests were conducted with the old version of Ripples running online while the new version was running locally on a computer. Both versions were logged in an account with admin permissions, meaning that the users were viewing the admin's view (similar to an operator's view)

5.2.1 Test Scenario

The test scenario was composed of the following tasks:

- First task - Draw a plan similar to the one being shown in this image (Figure 5.1);
- Second task - Determine if the plan was drawn in a high AIS density area;
- Third task - Indicate the following values:
 - Sea Surface Temperature in the following locations: Matosinhos, Setúbal, and Faro (Figure 5.2);
 - Sea Level Anomaly in the Berlengas Islands (Figure 5.3);
 - Wind intensity and direction in Espinho (Figure 5.4);
- Fourth task - Returning to the plan, indicate if there is any AIS Ship whose predicted position in the following ten minutes is crossing the plan;
- Fifth task - Indicate the latitude, longitude, and orientation of the AIS Ship that is nearest to the plan;
- Sixth task - The LAUV that is closest to the plan (Figure 5.5) was executing this plan, however it had to abort it and is now resurfacing in this position. Indicate where the LAUV will be dragged to.

During the tests, we gathered information about the task's success, the user's struggles and their feedback.

Most users failed to read the value of sea level anomaly when using the old version; some identified that the scale did not correlate with the scale's colour values. Two users were able to indicate the Sea Level Anomaly value, but the value was wrong, as their interpretation of the scale was wrong. Two users struggled to read the Sea Level Anomaly value for the new version due to the opacity levels. Due to the feedback from the users, the opacity of the Sea Level Anomaly layer was increased after the tests.

Three users had difficulties interpreting the layer's legend units, and two users could not read the layer's legend values when using the old version.

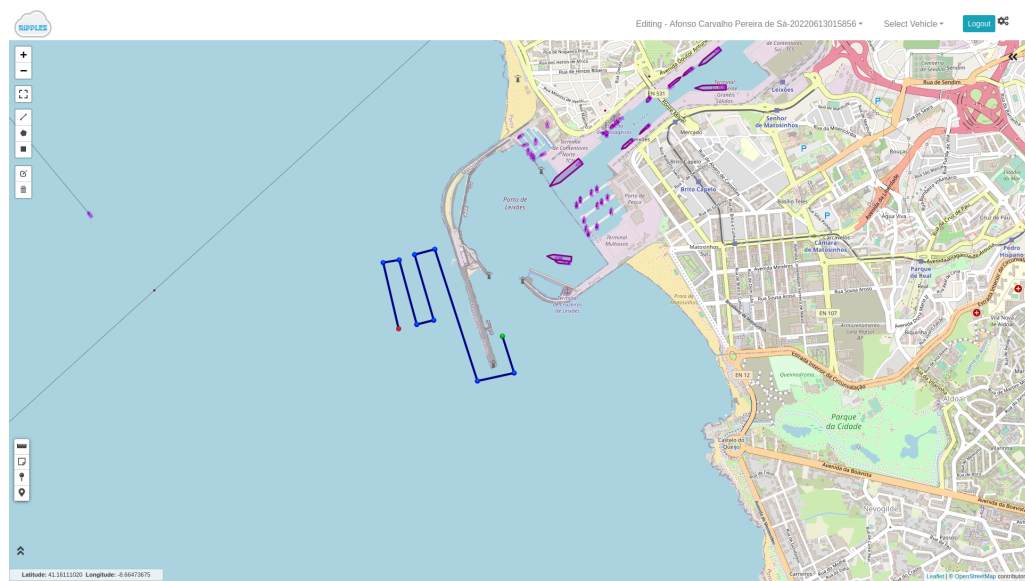


Figure 5.1: Reference plan

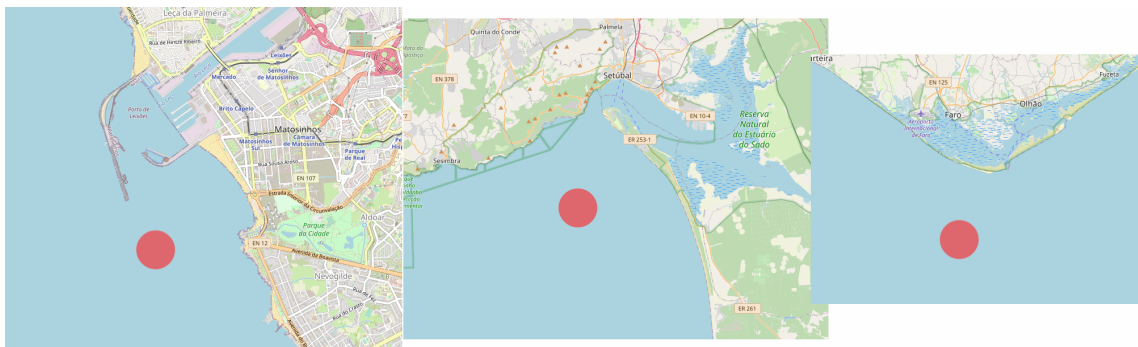


Figure 5.2: Locations to read the Sea Surface Temperature

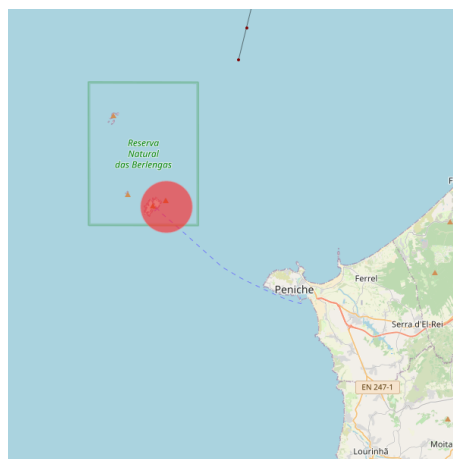


Figure 5.3: Location to read the Sea Level Anomaly

One user was able to replicate the bug described in Section 4.1.4, after indicating the sea surface temperature values, the user selected the Sea Level Anomaly layer without turning the

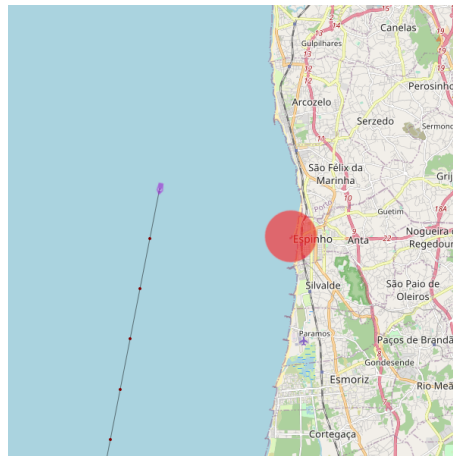


Figure 5.4: Location to read the Wind intensity and direction



Figure 5.5: Location of resurfaced LAUV

previous layer off. They then turned the temperature layer off, and all legends were removed from the screen, making it impossible to read the value of the Sea Level Anomaly layer. The user had to turn this layer on and off to show the legend again.

Only one user was not able to indicate correctly if an AIS Ship was within the operating window of the plan. This error occurred in the new version.

We also received feedback from two users that the sidebar toggle button was hard to spot when the map behind it showed a similar colour. This feedback led to an increase in the button's size; this change was done after performing the tests.

5.3 Results

In this section, we will discuss the results obtained. Firstly, we will discuss the workload values for the old version of Ripples and the new and will compare them. Secondly, we will discuss and compare the results for situational awareness.

5.3.1 Workload

One of the performance metrics being measured is the workload value; these values were measured using the NASA TLX rating system (Section 2.4.1). The objective of the new interface is to reduce the workload an operator feels when performing a specific task. As such, we measured the workload values after the user performed the tasks presented in the section 5.2.1. After finishing the tasks, the user was presented with the NASA TLX rating system, which can be consulted in the appendix B.

Both interface versions were tested in the same conditions and with the same rating system. As it was previously described in section 5.2, we took preventive measures to mitigate any bias by rotating which version the users were tested with first. The NASA TLX rating system is comprised of two parts first, a rating of six subscales: Mental Demand; Physical Demand; Temporal Demand; Performance; Effort; Frustration Levels, and a second part where the user is asked to do a series of pair-wise comparisons of what scale was the most important to their experience of workload. This will allow for the calculation of a weighted average. Here are the results we obtained:

Workload weights Old vs New Interface

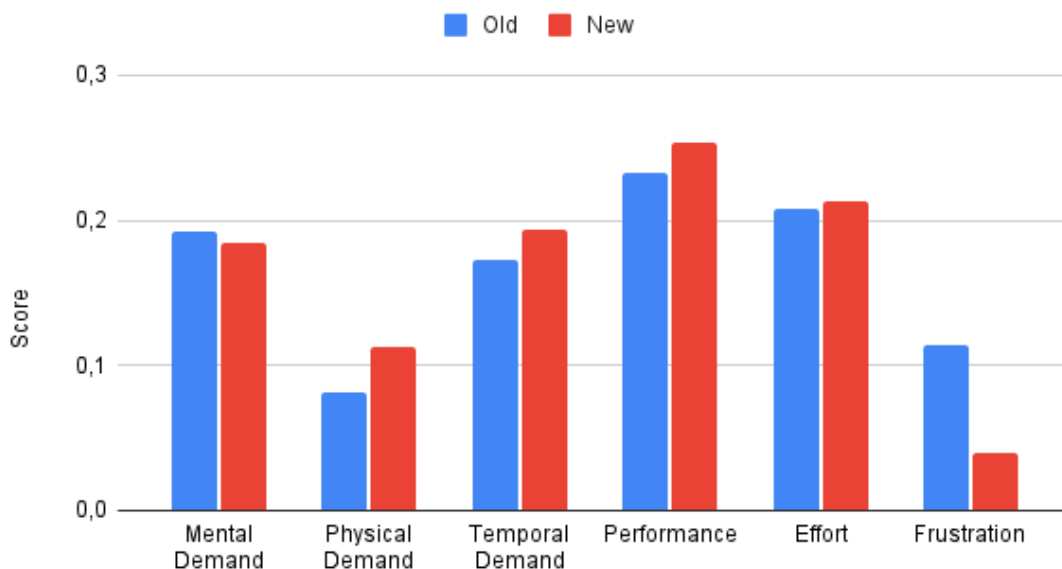


Figure 5.6: NASA TLX - Scale weights old vs new

Firstly, we will interpret the values of the scale's weights. It is observable that the physical demand had the least impact on the user's workload for the previous version, whereas, in the current version, the frustration was, by far, the least influential. For both versions, the performance had the most significant impact on the user's perception of workload. When comparing the weights in the graph of Figure 5.6 we can conclude that the weight given to **Frustration** reduced considerably, by 65%. The weight given to the **Physical Demand** had a slight increase, of 40%, and so did the **Temporal Demand** and **Performance**, by roughly 11% and 10% respectively. The other dimensions had negligible variation.

Workload weighted scores Old vs New Interface

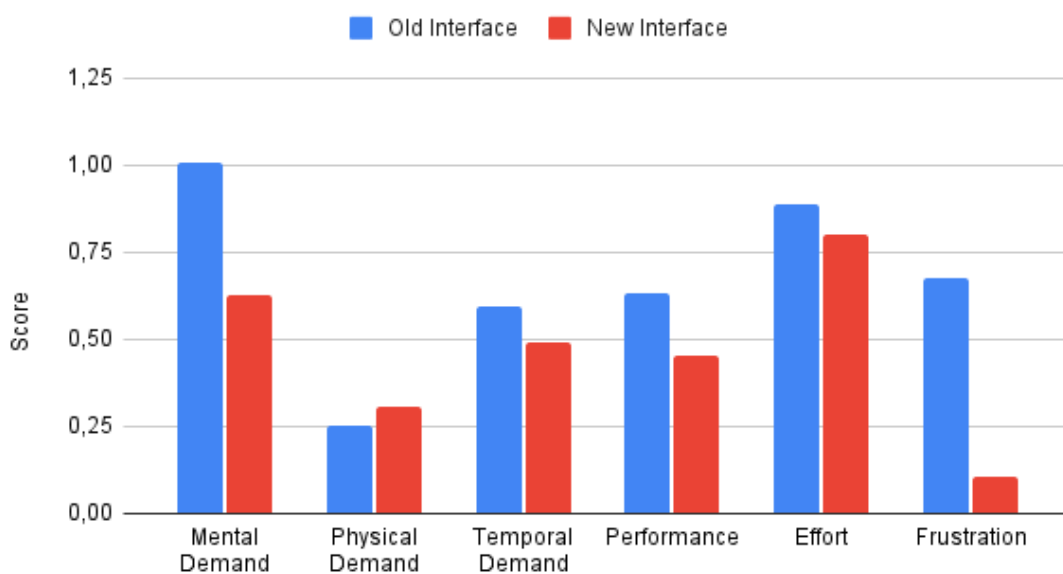


Figure 5.7: NASA TLX - Weighted scores old vs new

Regarding each scale's weighted scores, we turn to the graph in Figure 5.7, where it is observable that the values of **Mental Demand** and **Frustration** had a significant decrease, having reduced by approximately 38% and 28% respectively. However, it was the value of frustration that suffered the most significant decrease. It reduced to around 85% of the old interface's value. The value for the **Physical Demand** was the only one that suffered an increase; conversely, the performance value decreased in the new interface.

When exploring the average workload scores, we find that the new interface has a lower workload value, having reduced the workload by approximately 31% when compared to the original version. One person experienced a greater workload when using the new interface. We can conclude that we were able to reduce the operator's workload when using Ripples. This was possible due to a considerable reduction in frustration levels and the mental demand to perform a task. We present statistical analysis to further confirm the statistical significance of these results in section 5.4.

	Old Interface	New Interface	Difference
	Workload (0 - 10)	Workload (0 - 10)	Workload (0 - 10)
Test Subject 1	2,200	1,930	-0,270
Test Subject 2	3,530	0,930	-2,600
Test Subject 3	3,530	4,130	0,600
Test Subject 4	6,870	4,600	-2,270
Test Subject 5	5,330	3,270	-2,060
Test Subject 6	6,670	4,530	-2,140
Test Subject 7	3,200	1,200	-2,000
Test Subject 8	2,730	1,930	-0,800
Test Subject 9	2,670	2,600	-0,070
Test Subject 10	3,730	2,800	-0,930
Average	4,046	2,792	-1,254

Table 5.1: Workload scores

Individual Workload Scores

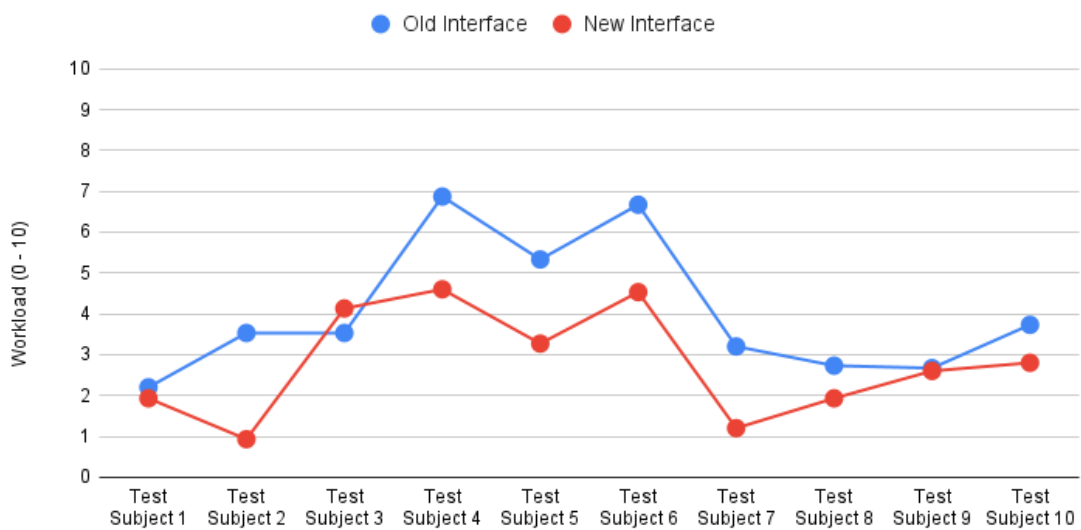


Figure 5.8: Individual workload scores

5.3.2 Situational Awareness

The other performance metric being measured is the value of situational awareness of the operator; these values were measured using the SART rating system (Section 2.3.2). The objective of the new interface is to increase the situational awareness of an operator when performing a specific task. As such, we measured the situational awareness values after the user performed the tasks presented in the section 5.2.1. After finishing the tasks, the user was presented with the SART rating system, which can be consulted in the appendix C. Both versions of the interface were

tested in the same conditions and same rating system. Here are the results we obtained:

Situation Awareness Scores Old vs New Interface

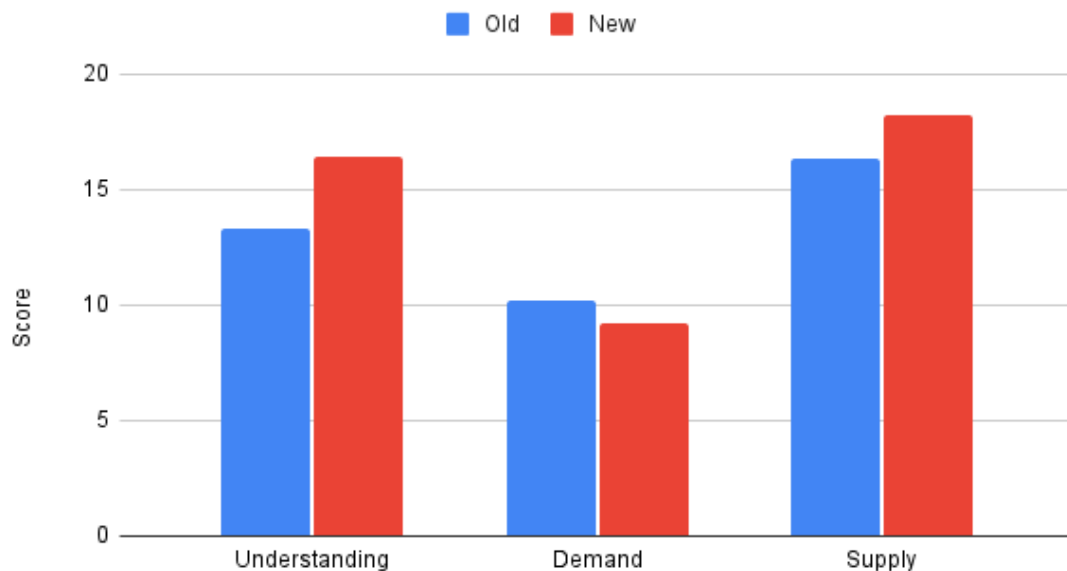


Figure 5.9: SA scores

In order to better understand how each scale of this rating system affects the situation awareness score, we will show the formula to obtain this value: $SA = U + S - D$, where U is the understanding value, S is the supply, and D is the demand. An increase in Understanding (U) and Supply (S) and a decrease in Demand (D) results in an increase in situational awareness (SA).

Looking at Figure 5.9 we can see that there was an increase in understanding and supply and a decrease in demand. The most significant increase was with understanding, with an increase of approximately 23%. The supply increased by nearly 12%, and the demand decreased by 10%.

	Old Interface	New Interface	Difference
	SA (0 - 1)	SA (0 - 1)	SA (0 - 1)
Test Subject 1	0,500	0,583	0,083
Test Subject 2	0,600	0,833	0,233
Test Subject 3	0,633	0,650	0,017
Test Subject 4	0,400	0,667	0,267
Test Subject 5	0,433	0,550	0,117
Test Subject 6	0,483	0,517	0,033
Test Subject 7	0,517	0,583	0,067
Test Subject 8	0,617	0,733	0,117
Test Subject 9	0,683	0,633	-0,050
Test Subject 10	0,700	0,817	0,117
Average	0,557	0,657	0,100

Table 5.2: Situational Awareness scores

When analysing the Table 5.2 we can conclude that only one user had a lower value of situational awareness when using the new interface. The average SA score for the new interface had an increase of approximately 18% to the value of the old interface. It is possible to arrive at the conclusion that the new version was capable of increasing the operator's situational awareness; this was possible by an increase in the understanding and supply of the information as well as a reduction of the demand.

Individual Situational Awareness Scores

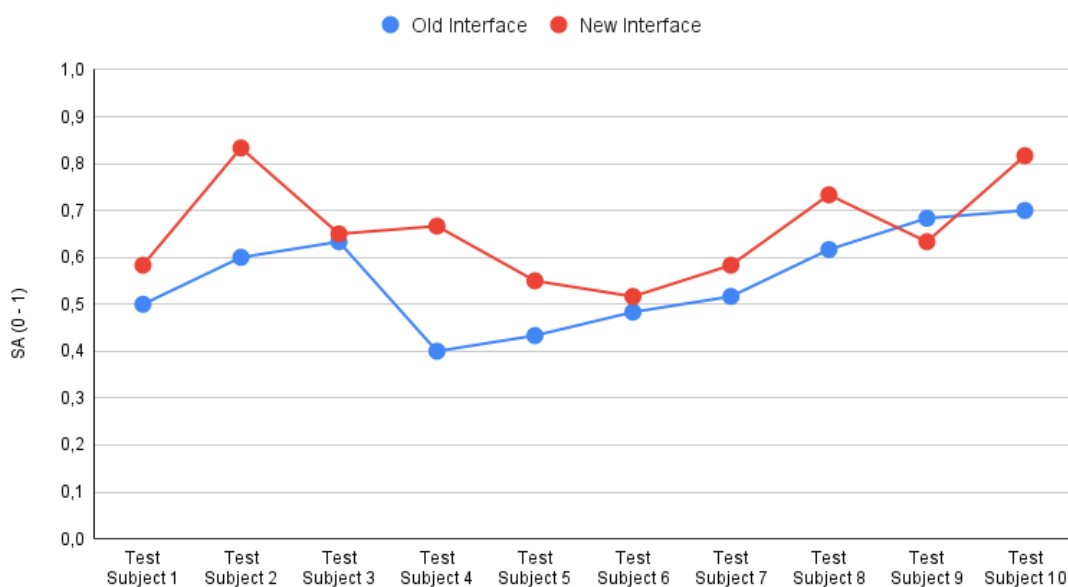


Figure 5.10: Individual SA scores

5.4 Discussion

The tests were limited to the capabilities of Ripples, we intended to add more variables, such as simulated vehicles executing plans in order to add more possible distractions and expand the tasks. One other point that could have been improved in these tests would be a task pool, comprised of similar tasks, from which we could rotate, in order not to have the users repeat the same tasks between versions.

A statistical analysis was done to determine if the increased situational awareness and decreased workload values compared to the old interface were statistically significant. The workload and situational awareness measurements are pairs of before and after values of the same test subjects. The test used in this scenario is the paired sample t-test; it is used to determine whether the mean change for pairs of values of the same group is significantly different from zero (null hypothesis).

	Old Interface		New Interface		Difference	
	Workload (0 - 10)	SA (0 - 1)	Workload (0 - 10)	SA (0 - 1)	Workload (0 - 10)	SA (0 - 1)
Test Subject 1	2,200	0,500	1,930	0,583	-0,270	0,083
Test Subject 2	3,530	0,600	0,930	0,833	-2,600	0,233
Test Subject 3	3,530	0,633	4,130	0,650	0,600	0,017
Test Subject 4	6,870	0,400	4,600	0,667	-2,270	0,267
Test Subject 5	5,330	0,433	3,270	0,550	-2,060	0,117
Test Subject 6	6,670	0,483	4,530	0,517	-2,140	0,033
Test Subject 7	3,200	0,517	1,200	0,583	-2,000	0,067
Test Subject 8	2,730	0,617	1,930	0,733	-0,800	0,117
Test Subject 9	2,670	0,683	2,600	0,633	-0,070	-0,050
Test Subject 10	3,730	0,700	2,800	0,817	-0,930	0,117
Average	4,046	0,557	2,792	0,657	-1,254	0,100

Table 5.3: Overall results

Regarding workload, the t-test hypothesis was a lower-tailed alternative hypothesis, meaning we were testing if the values of workload of the previous version were greater than the current version. Results of the paired-t test indicated that there is a significantly large difference between the workload values of the old interface ($M = 4$, $SD = 1.7$) and the new interface ($M = 2.8$, $SD = 1.3$), $t(9) = 3.6$, $p = 0.003$.

For situational awareness, the t-test hypothesis was an upper-tailed alternative hypothesis, meaning we were testing if the values of situation awareness of the previous version were smaller than the current version. Results of the paired t-test indicated that there is a significant large difference between the situation awareness values of the old interface ($M = 0.6$, $SD = 0.1$) and the values of the new interface ($M = 0.7$, $SD = 0.1$), $t(9) = 3.3$, $p = 0.004$.

The results are statistically significant and satisfactory. It is possible to conclude that the tests were successful in showing that the main objectives were achieved, having **reduced workload** by approximately **31%** and **increased situational awareness** values by **18%**.

Chapter 6

Conclusions

This last chapter presents the conclusions of the work done, in it there is a summarised overview of the work, a discussion of the objectives and their completion, and finally there is a presentation of potential ideas to explore as future work.

6.1 Summary of the Work Done

This dissertation had the high-level objective of improving Ripples' interface, as such it was expected that we solved the underlying problems with the interface, in order to improve the end-user's performance when using Ripples. Ripples is used for planning campaigns and execution control, where the decision-making process will have real-world effects to valuable assets, such as the LSTS' LAUV. Consequently, it was mandatory that any changes to the interface came with the understanding that they must aid with decision-making and problem-solving. We defined the goals of maintaining or reducing the operator's workload and maintaining or increasing their situational awareness. In order to obtain these goals we followed these steps:

Firstly, having decided on our performance metrics of workload and SA, we researched methods of evaluating an operator's workload and situational awareness when realising a task. After analysing several methods for each metric we decided upon NASA TLX and SART as the tools to measure these variables. These two methods were chosen because of their practicality and ease of application as well as their compatibility.

Aftwards, we shifted our research to similar platforms to Ripples, with focus on data visualisation software and command and control. This research was important in order to figure what other people in this area have done and what methods they have implemented to mitigate problems that Ripples suffers from. For example, the CMEMS system aided with some design choices for Ripples.

The last part of our research process shifted towards design practices and maritime UI. The UI/UX area is broad, as such we narrowed the research to human-computer interaction, human-centered design, and decision support. By doing so we gathered several design practices and guidelines that informed the design process of Ripples. We also researched maritime UI and

found other interfaces designed with similar goals to the ones we set for our work. It was during this phase that we came across the term Ecological Interface Design. After some exploration of this design framework we found that it was the ideal framework to implement with Ripples.

EID states that there must be an extensive understanding of the working domain of the system to guide the interface design. Consequently, we did an analysis of the working domain of the system as well as the users requirements for it. This process helped us understand how Ripples worked and what was expected out of it. It was also helpful in gathering feedback and frustrations the users had when using Ripples.

It was now time to start the development process, during this time we developed several iterations of Ripples. Working with Ripples proved to be a challenge due to the constraints of having to work with a very restricting version of react-leaflet and software with multiple dependencies that prevented upgrading the react-leaflet library. Nevertheless, the final iteration of Ripples tackled most of the problems we had identified.

With the development of the new interface and all the new functionalities it was time to test both versions of Ripples and obtain their values or workload and situation awareness. We tested the members of the laboratory, asking them to perform a set of specific tasks and applying the NASA TLX and SART techniques afterwards. Once the tests were concluded we compared the values and performed a statistical analysis to validate the results.

6.2 Objectives Completion

Revisiting the initial objectives we set for this dissertation in Section 1.5, with the knowledge of the results obtained, it is possible to conclude that all objectives were fulfilled. The **workload** values of users were **reduced**, by 31%, when using the new interface; The value of **situational awareness** experienced by the users **increased**, by 18%; The **entropy** between users was **reduced** with the introduction of role-specific views; The **information** being displayed **increased** and so did the fidelity of this information.

The work done during this dissertation was presented at "Encontro de Oceanografos 2022" as a poster entitled "Ripples: Increasing human supervision over operational assets for a persistent and efficient ocean observation". At the time of the presentation the results were not yet available, as such the presentation was focused on Ripples and the development of the new interface. The poster can be consulted in the Appendix E.

6.3 Future Work

There is still work to be done with Ripples; the current version has tackled most of the problems with the original interface. Nevertheless, there are still some issues that can still be solved and features that can be added. The following ideas are some of the future work that we identified:

Add automatic scale adjustment; this feature would be simple to implement if this it was accessible via the Copernicus API. One possibility would be to request the maintainers of this API for this feature. However, if that does not work, it would be possible to re-request the rendered chunks and apply the previously identified method of averaging the RGB values. This last idea would have some performance issues as each time the layer is rendered, it would double the network requests and then calculate the average of the RGB values.

Study/Change the iconography; some icons currently represent different types of assets, so it would be beneficial to add icons to represent the different assets. Other icons can be hard to spot or difficult to interpret. Changing these icons would also benefit Ripples. Moreover, it would be possible to give more information to the users with some icons. For example, LAUV icons could incorporate red, orange, or green outlines to show their status.

Better way of merging information when showing multiple layers; currently, the user can select multiple layers simultaneously, making the displayed information hard to read. We choose to leave the possibility to select multiple layers to leave it to the user's preference on how many layers they want active. Thus, it would be ideal if there was a better method of showing multiple layers at a time. We are going to present two solutions to this problem. The first idea would be to implement a feature similar to CMEMS's functionality of being able to change the layer's colour scale. The other idea is to change how some layer's information is being displayed. For example, the Sea Surface Velocity layer could be represented as vectors, their orientation indicates the direction, and their colour would indicate the velocity value.

Change bathymetry layer; during the development process we considered changing the current bathymetry layer to the navionics¹ bathymetry layer, due to its greater detail. As such, we contacted the Navionics support center, but due to time constraints we could not get this layer in time.

Upgrade the react-leaflet version; the current version of react-leaflet is highly restrictive and allows for no customisation. Upgrading the react-leaflet version to version three or higher would allow for customisation of leaflet elements.

¹<https://webapp.navionics.com/>

References

- [1] Kevin B. Bennett and John Flach. Ecological Interface Design: Thirty-Plus Years of Refinement, Progress, and Potential. *Human Factors*, 61(4):513–525, June 2019. Publisher: SAGE Publications Inc.
- [2] Kevin B. Bennett and John M. Flach. *Display and Interface Design: Subtle Science, Exact Art*. CRC Press, 0 edition, March 2011.
- [3] Clark Borst, John Flach, and Joost Ellerbroek. Beyond Ecological Interface Design: Lessons From Concerns and Misconceptions. *Human-Machine Systems, IEEE Transactions on*, 45:164–175, April 2015.
- [4] Brad Cain. A Review of the Mental Workload Literature. *English*, page 35, July 2007.
- [5] N. A. Costa, E. Holder, and S. N. MacKinnon. Implementing human centred design in the context of a graphical user interface redesign for ship manoeuvring. *International Journal of Human-Computer Studies*, 100:55–65, April 2017.
- [6] Paulo Sousa Dias, Maria Costa, José Pinto, Keila Lima, Luís Venâncio, Miguel Aguiar, and João Borges Sousa. Large Scale Unmanned Vehicles Oceanic Exercise REP(MUS)19 Field Report. In *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, pages 1–2, September 2020. ISSN: 2377-6536.
- [7] Mica Endsley. Direct Measurement of Situation Awareness: Validity and Use of SAGAT. *Situation Awareness: Analysis and Measurement*, January 2000.
- [8] Mica Endsley. Situation awareness analysis and measurement, chapter theoretical underpinnings of situation awareness. *A Critical Review*, pages 3–33, January 2000.
- [9] Mica R. Endsley. A Systematic Review and Meta-Analysis of Direct Objective Measures of Situation Awareness: A Comparison of SAGAT and SPAM. *Human Factors*, 63(1):124–150, February 2021. Publisher: SAGE Publications Inc.
- [10] Mica R. Endsley, Stephen J. Selcon, Thomas D. Hardiman, and Darryl G. Croft. A Comparative Analysis of Sagat and Sart for Evaluations of Situation Awareness. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 42(1):82–86, October 1998.
- [11] Rafael Falcon, Rami Abielmona, Benjamin Desjardins, and Emil Petriu. Fuzzy/human risk analysis for maritime situational awareness and decision support. In *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6, July 2017. ISSN: 1558-4739.
- [12] D. Fay, N. Stanton, and A.P.J. Roberts. Exploring Ecological Interface Design for Future ROV Capabilities in Maritime Command and Control. In *AHFE 2018 International Conference on Human Factors in Transportation, 21-25 July 2018*, Advances in Human Aspects

- of Transportation. Proceedings of the AHFE 2018 International Conference on Human Factors in Transportation. *Advances in Intelligent Systems and Computing* (786), pages 264–73, Cham, Switzerland, 2019. Springer International Publishing.
- [13] António Sérgio Ferreira, José Pinto, Paulo Dias, and João Borges de Sousa. The LSTS software toolchain for persistent maritime operations applied through vehicular ad-hoc networks. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 609–616, June 2017.
 - [14] Kevin Gomes, Danelle Cline, Duane Edgington, Michael Godin, Thom Maughan, Mike McCann, Tom O'Reilly, F. Bahr, F. Chavez, Monique Messié, Jnaneshwar Das, and Kanna Rajan. ODSS: A decision support system for ocean exploration. pages 200–211, April 2013.
 - [15] Rui Gonçalves, António Ferreira, Jose Pinto, João Sousa, and Gil Gonçalves. Authority Sharing in Mixed Initiative Control of Multiple Uninhabited Aerial Vehicles. pages 530–539, July 2011.
 - [16] Sandra G Hart and Moffett Field. NASA-TASK LOAD INDEX (NASA-TLX); 20 YEARS LATER. page 5.
 - [17] Sandra G. Hart and Lowell E. Staveland. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In Peter A. Hancock and Najmedin Meshkati, editors, *Advances in Psychology*, volume 52 of *Human Mental Workload*, pages 139–183. North-Holland, January 1988.
 - [18] Kai Hormann and Alexander Agathos. The point in polygon problem for arbitrary polygons. *Computational Geometry*, 20(3):131–144, November 2001.
 - [19] Michele Jacobi, Rob Braswell, Amy Merten, Nancy Kinner, and Kurt Schwehr. Environmental Response Management Application (ERMA) - Web-based GIS Data Display and Management System for Oil Spill Planning and Environmental Response. *Coastal Geotools 2009*, March 2009.
 - [20] Jeff Johnson. *Designing with the mind in mind: simple guide to understanding user interface design rules*. Morgan Kaufmann Publishers/Elsevier, Amsterdam ; Boston, 2010. OCLC: ocn308185111.
 - [21] Irvin Fang Jau Lim. Comprehensive maritime domain awareness : an idea whose time has come. Working Paper, 2007. Accepted: 2009-02-05T09:32:26Z.
 - [22] Somaiyeh MahmoudZadeh, David M.W Powers, Karl Sammut, Adham Atyabi, and Amirmehdi Yazdani. A hierarchal planning framework for AUV mission management in a spatiotemporal varying ocean. *Computers & Electrical Engineering*, 67:741–760, April 2018.
 - [23] Yemao Man, Reto Weber, Johan Cimbritz, Monica Lundh, and Scott N. MacKinnon. Human factor issues during remote ship monitoring tasks: An ecological lesson for system design in a distributed context. *International Journal of Industrial Ergonomics*, 68:231–244, November 2018.
 - [24] K Nordby, S Frydenberg, and J Fauske. DEMONSTRATING A MARITIME DESIGN SYSTEM FOR REALISING CONSISTENT DESIGN OF MULTI-VENDOR SHIP'S BRIDGES. *Human Factors*, page 10, 2018.

- [25] U.S. Navy Office of Information. United States Participates in Exercise REP(MUS) 2021. <https://www.navy.mil/Press-Office/News-Stories/Article/2801088/united-states-participates-in-exercise-repmus-2021/>.
- [26] Glenn Osga, Mike McWilliams, Darren Powell, David Kellmeyer, Jerry Kaiwi, and Al Ahumada. Unmanned Surface Vehicle Human-Computer Interface for Amphibious Operations. Technical report, SPACE AND NAVAL WARFARE SYSTEMS CENTER PACIFIC SAN DIEGO CA, August 2013. Section: Technical Reports.
- [27] José Pinto, Paulo Sousa Dias, and João Borges de Sousa. Ripples: a tool for supervision and control of remote assets. *Instrumentation Viewpoint*, (20):51–51, 2018. Accepted: 2019-01-08T13:22:21Z Publisher: SARTI.
- [28] HP repository. Situation Awareness Rating Technique (SART) | HP repository. <https://ext.eurocontrol.int/ehp/?q=node/1608>.
- [29] HP repository. Subjective Workload Assessment Technique (SWAT) | HP repository. <https://ext.eurocontrol.int/ehp/?q=node/1588>.
- [30] Robert. react-leaflet-custom-layer-control-ts, 2021. <https://github.com/Roschl/react-leaflet-custom-layer-control-ts>.
- [31] Copernicus Marine Service. MyOcean Viewer | CMEMS. <https://marine.copernicus.eu/access-data/myocean-viewer>.
- [32] Ben Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Longman Publishing Co., Inc., USA, 3rd edition, 1997.
- [33] Daniel Sunday. *Practical Geometry Algorithms: With C++ Code*. Amazon Digital Services LLC - KDP Print US, May 2021.
- [34] K.J. Vicente and J. Rasmussen. Ecological interface design: theoretical foundations. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(4):589–606, July 1992. Conference Name: IEEE Transactions on Systems, Man, and Cybernetics.
- [35] J. Wang, H. S. Sii, J. B. Yang, A. Pillay, D. Yu, J. Liu, E. Maistralis, and A. Saajedi. Use of Advances in Technology for Maritime Risk Assessment. *Risk Analysis*, 24(4):1041–1063, 2004. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.0272-4332.2004.00506.x>.
- [36] Guangjiang Wu, Yiqian Wu, Xi Lu, Shenghang Xu, and Chuan Wang. Human–Machine Interface Optimization Design Based on Ecological Interface Design (EID) Theory. In Shengzhao Long and Balbir S. Dhillon, editors, *Man-Machine-Environment System Engineering*, Lecture Notes in Electrical Engineering, pages 715–723, Singapore, 2020. Springer.

Appendix A

Users Questionnaire



Perguntas

Respostas

4

Definições

4 respostas



Aceitar respostas



Resumo

Pergunta

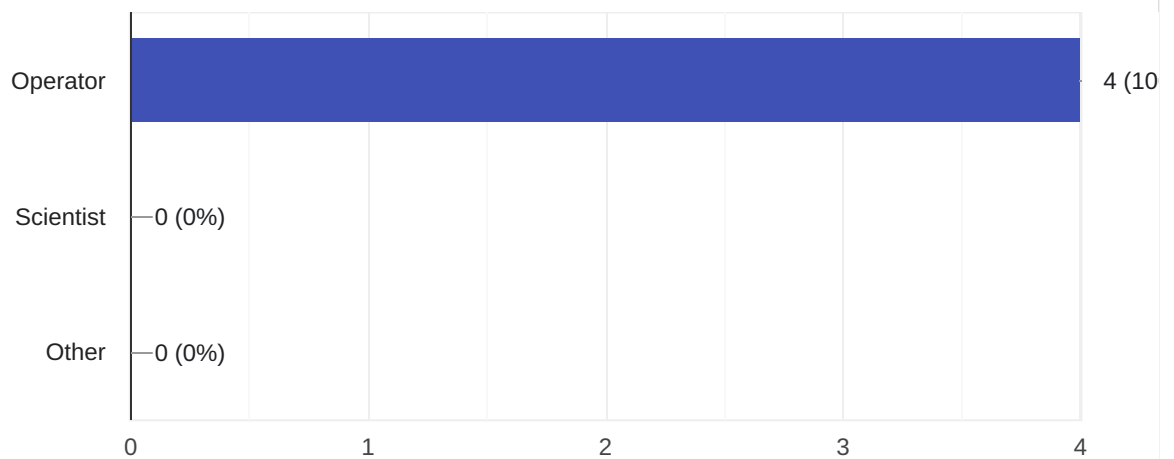
Individual

What is your role when conducting maritime operations?



Copiar

4 respostas

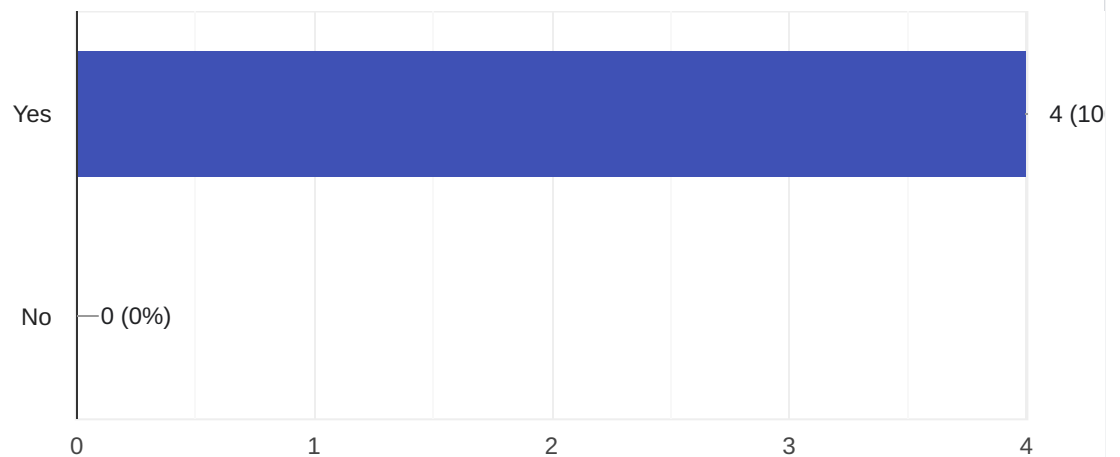


Have you used autonomous/unmanned vehicles/systems when performing maritime operations before?



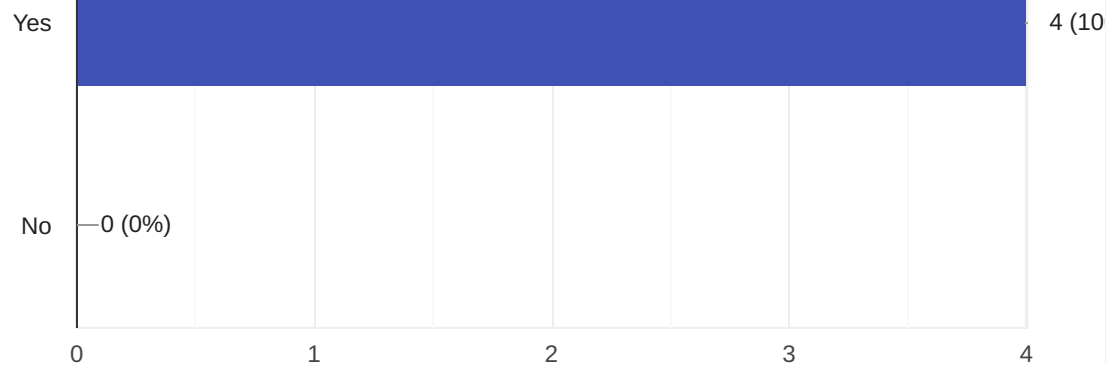
Copiar

4 respostas





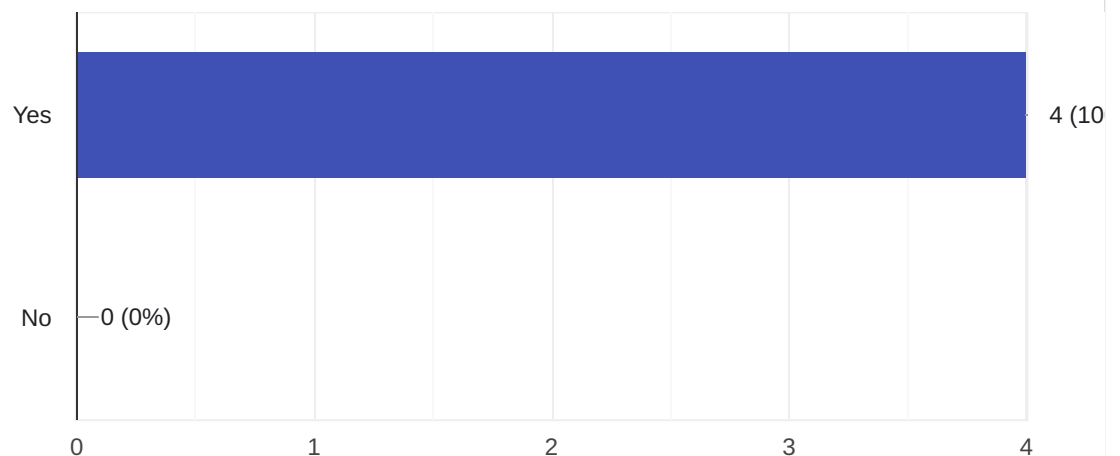
Perguntas Respostas 4 Definições



Have you ever used any type of software dashboard tool/interface to help supervise and/or monitor your maritime operations/campaigns?



4 respostas



List the most relevant assets (sensor, vehicles, models, etc..) you deploy/use in your maritime operation campaigns regularly.

4 respostas

vehicles, communication gateways, vessels

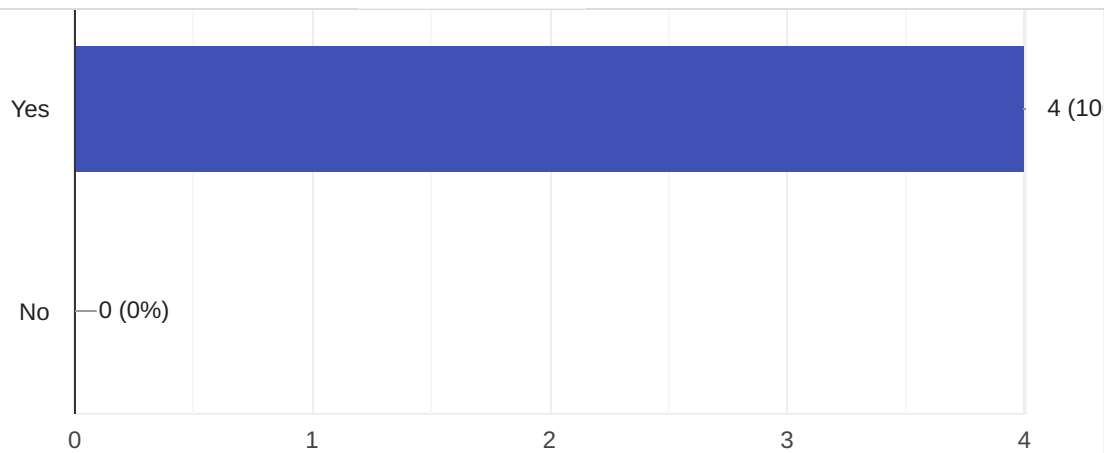
Vehicles

vehicles

AUVs (both for oceanographic and bottom mapping purposes)



Perguntas Respostas 4 Definições



If NOT, would this possibility be important/crucial?

0 respostas

Ainda não existem respostas a esta pergunta.

Please list (up to a maximum of 6) mission moments/situations where having the ability to supervise and/or monitor all said assets in a centralized manner would be beneficial.

3 respostas

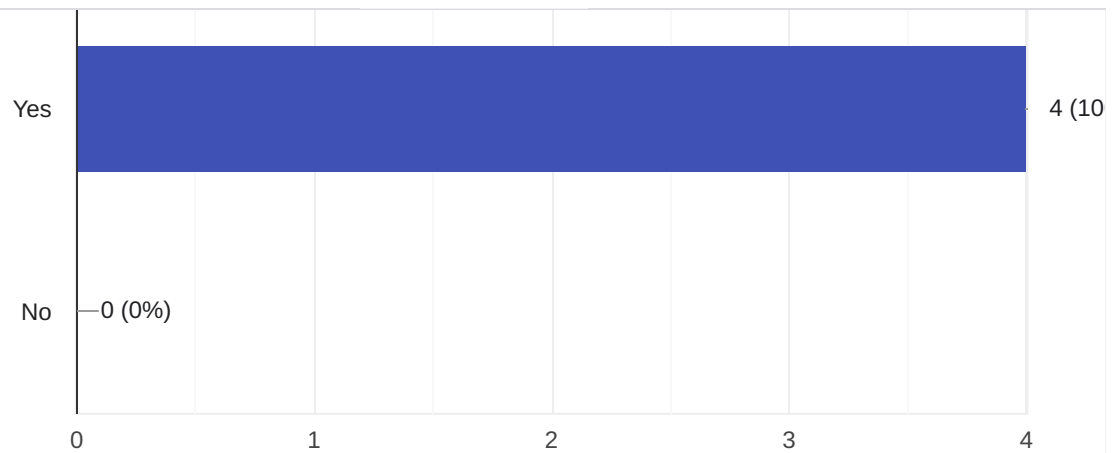
When third parties... either vessels or unmanned systems interact/join the operational scenario; When the assets are performing a joint collaborative task.

Collisions and heavy maritime traffic.

If a large area is to be covered, having this ability is quite important cost-effective wise. Also in complex environments that require multiple assets it's vital to have a synoptic view / control of the complete theater of operations to guarantee the safety of all assets and help "de-conflict" potential issues (for instance, if one asset has a problem you can easily replan or adjust other assets to adapt to that situation).



Perguntas Respostas 4 Definições



Please list (up to a maximum of 6) pieces of data/information that fall under the previous relevance category.

4 respostas

Sidescan data; Image data; CTD; environmental data

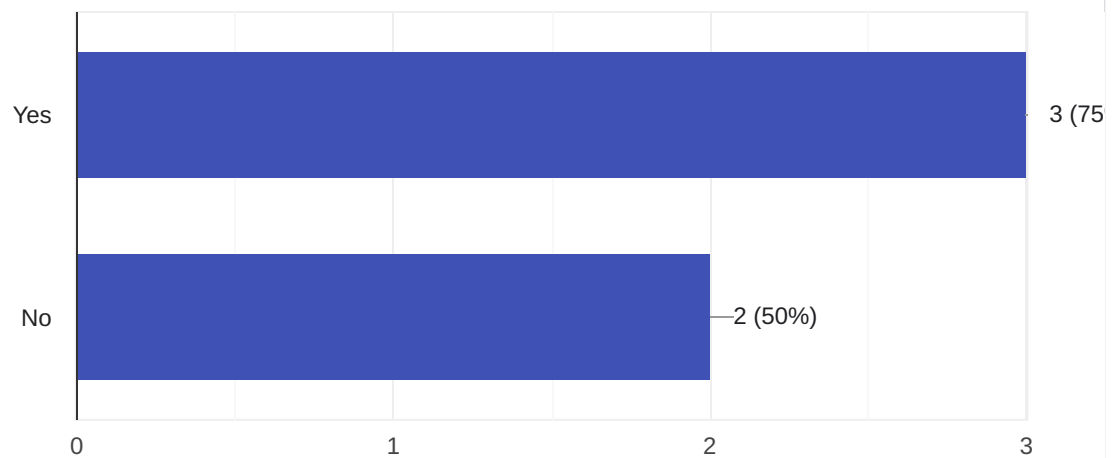
Position, speed and heading.

status and position

In oceanographic scenarios it's quite common to sample the area, review the data and then decide where to go next for instance. It may also be valuable to be able to review quickly in the end of a survey if the quality of the collected data is good or if adjustments may be needed to the payloads / plan to improve said quality. In a scenario where you're searching for something, review the data after a larger survey to pinpoint points of interest where to do close-up (more detailed) surveys may also be quite valuable to make the mission more efficient.



Perguntas Respostas 4 Definições



If NOT, please list (up to a maximum of 6) constraints that, normally, reduce the quick accessibility of those pieces of data/information.

2 respostas

communication channel; data bandwidth.

iridium

Please list (up to a maximum of 6) mission plans/decisions that can be impacted by being able to access/analyze these pieces of data/information in a small enough timeframe .

4 respostas

reduce the length of the plan; send the asset to a secure location

Safe plans/areas.

avoid collisions

In case of oceanography it may impact if the goal of the mission is accomplished in time or not.



Perguntas **Respostas** 4 Definições

coordination of several teams (joint demonstrations); have situational awareness of atmospheric and underwater conditions;

Recovers vehicle from water when it is not operational.

Make sure the data collected during the mission is being done properly (sampling method, when and where, format, correlation to other important variables, etc.).

Which of these difficulties, if any, are related to the software tools used in the campaign life cycle?

2 respostas

all

A lot of this can be solved with proper planning with the "right" people for the application. In terms of software, tools such as annotations on the map, plotting options or layers to plot important information to the mission on the map may be quite helpful in this. But of course the mission requirements need to be studied beforehand, so the software team can go through them and analyze if developments are required for the mission, in order to make sure the data collected will be useful.

Ripples

Could you list some of the functionalities of Ripples that you use in the context of your maritime operations?

4 respostas

information layers; collision prediction/avoidance

Check payload data and position of vehicles

locations, weather and AIS layers

Iridium/SPOT communications over ripples server. It's specially useful to display the synoptic view of the operations with AIS traffic to partners involved in an exercise, since it can run in a browser in any platform.



Perguntas **Respostas** 4 Definições

can't tell

With a lot of data to be shown, navigation in Ripples is very slow/heavy

Number of icons/assets

Distinguishing what is happening at the moment and from what has happened already.

Could you list some difficulties when using this software that are impactful in terms of their effects on the smoothness of operations [that makes the operation experience more cumbersome when performing any task]?

3 respostas

Frequency of assets update location/information;

Number of icons/assets

If a lot of layers are selected it becomes quite heavy to run in some machines.

Looking at the points above, from your point of view, what would you consider the biggest User Interface-related feature (functional or non-functional) that could help solve or mitigate this reality?

3 respostas

having a better communication relay; having a better communication interface (better feedback from the communication gateway)

Group, hide or redraw some assets

Make it lighter in terms of resources consumption (if possible) and clean-up the sometimes "excessive visual clutter". Make what is coming through in "real-time" more evident and discard (gray out or hidden layers by option perhaps?) what has already happened so it's easy for the user to perceive the relevant information.



Perguntas

Respostas 4

Definições

can't tell

Wi-Fi connection

Improve specifications of Ripples server to account for scalability may be the first thing that comes to mind for me.

Appendix B

NASA TLX

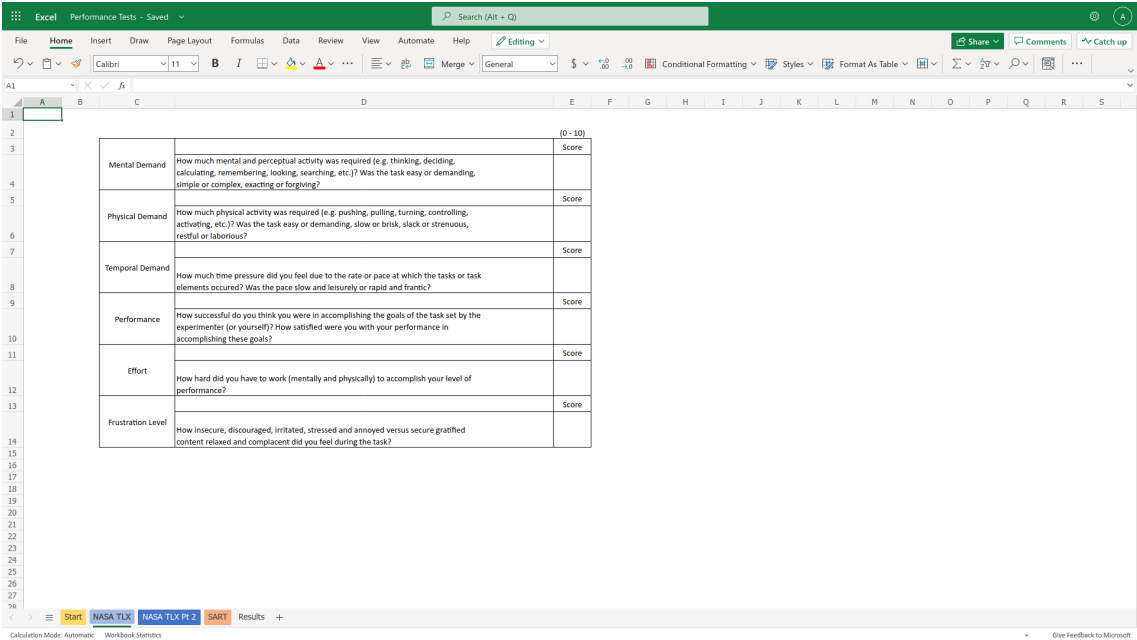


Figure B.1: NASA TLX - Scoring

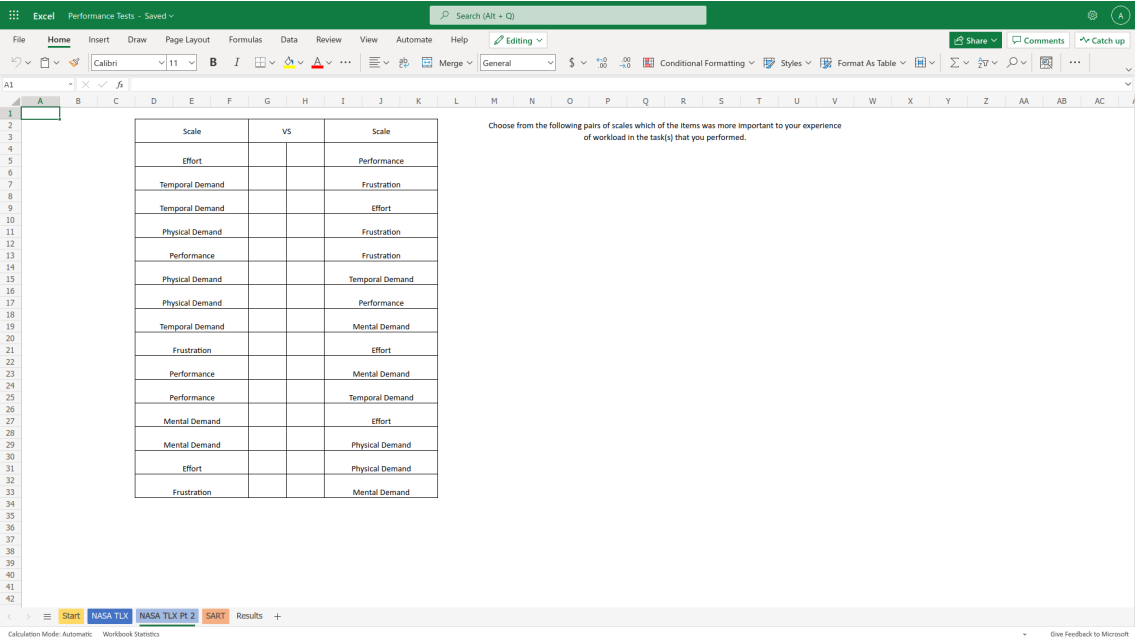


Figure B.2: NASA TLX - Weights

Appendix C

SART

Excel Performance Tests - Saved

Search (Alt + Q)

File Home Insert Draw Page Layout Formulas Data Review View Automate Help Editing

Calibri 11 B I Merge General \$ +0.00 -0.00 Conditional Formatting Styles Format As Table

C42

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1			Score (1 - 7)													
2		Instability of situation														
3		How changeable is the situation? Is the situation highly unstable and likely to change suddenly (High) or is it very stable and straightforward (Low)?														
4																
5		Complexity of situation														
6																
7		How complicated is the situation? Is it complex with many interrelated components (High) or is it simple and straightforward (Low)?														
8																
9		Variability of situation														
10																
11		How many variables are changing within the situation? Are there a large number of factor varying (High) or are there very few variables changing (Low)?														
12																
13		Arousal														
14																
15		How aroused are you in the situation? Are you alert and ready for activity (High) or do you have a low degree of alertness?														
16																
17		Spare mental capacity														
18																
19		How much mental capacity do you have to spare in the situation? Do you have sufficient to attend to many variables (High) or nothing to spare at all (Low)?														
20																
21		Concentration														
22																
23		How much are you concentrating on the situation? Are you concentrating on many aspects of the situation (High) or focussed on only one (Low)?														
24																
25		Division of attention														
26																
27		How much is your attention divided in the situation? Are you concentrating on many aspects of the situation (High) or focussed on only one (Low)?														
28																
29		Information Quantity														
30																
31		How much information have you gained about the situation? Have you received and understood a great deal of knowledge (High) or very little (Low)?														
32																
33		Information Quality														
34																
35		How good is the information you have gained about the situation? Is the knowledge communicated very useful (High) or is it insufficient (Low)?														
36																
37		Familiarity with situation														
38																
39		How familiar are you with the situation? Do you have a great deal of relevant experience (High) or is it a new situation (Low)?														
40																
41																
42																

Calculation Mode: Automatic Workbook Statistics

Start NASA TLX NASA TLX PL 2 SART Results

Give Feedback to Microsoft

Figure C.1: SART

Appendix D

Workload and SA scores

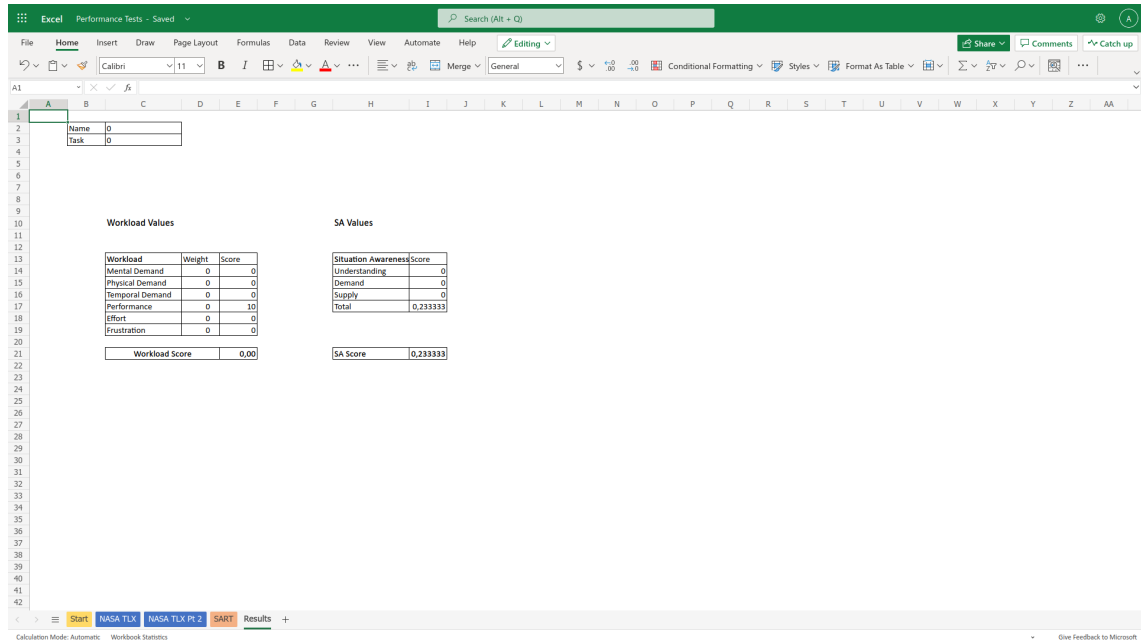


Figure D.1: Workload and SA Scores

Appendix E

Presentation at Encontro de Oceanografia 2022

Is it possible to increase human supervision by applying design techniques to Ripples' interface?

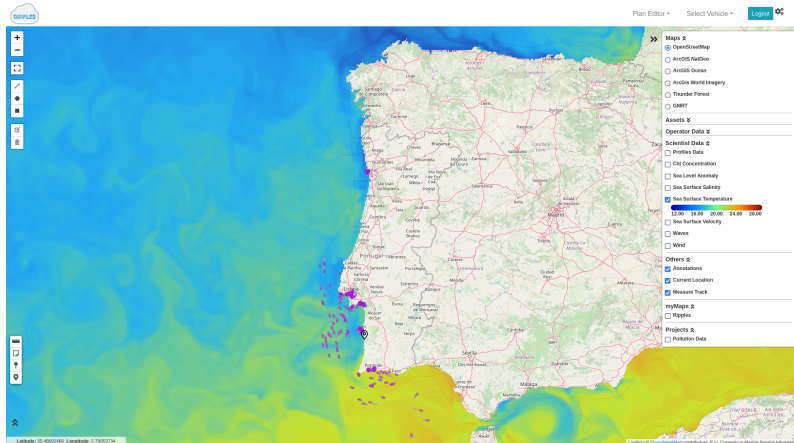


Figure 1. Ripples - Sea Surface Temperature

Ripples

Ripples is a communications hub for **data dissemination** and **situation awareness** in operations at sea, including oceanographic campaigns using traditional and new robotic and autonomous observation systems. It is a web-based service that is integrated within the LSTS toolchain.

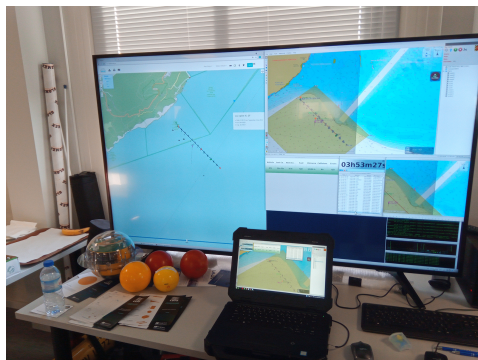


Figure 2. REP(MUS) 21 - Ripples in use

Ripples is a secure, persistent, and collaborative maritime data processing and analysis service aimed at:

- high-level decision-making;
- situational awareness

This software is continuously under development to achieve the goal of being capable of Multisystem Command and Control (C2).

Increasing human supervision

Ripples is used in remote environment missions where it is necessary to make critical decisions, thus, a redesign of its interface was done in order to **increase situation awareness** and **reduce workload**.

Ecological Interface Design

Ripple's redesign was done by applying design techniques, among which Ecological Interface Design (EID) stood as the most relevant.

EID is an interface design methodology that prioritises reducing mental workload and supporting knowledge-based reasoning. The following principles were applied when redesigning Ripple's interface:

- The interface design shall be **user-oriented**;
- The interface shall provide enough **feedback** and **appropriate information**;
- The mapping from information to interface shall be **consistent** and **unique**;
- The interface design shall adopt **different levels of abstraction** to provide information and support people's **decision-making activities**.

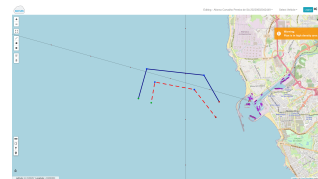


Figure 3. Ripples - plan creation feedback

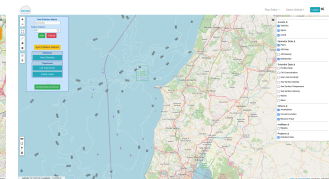


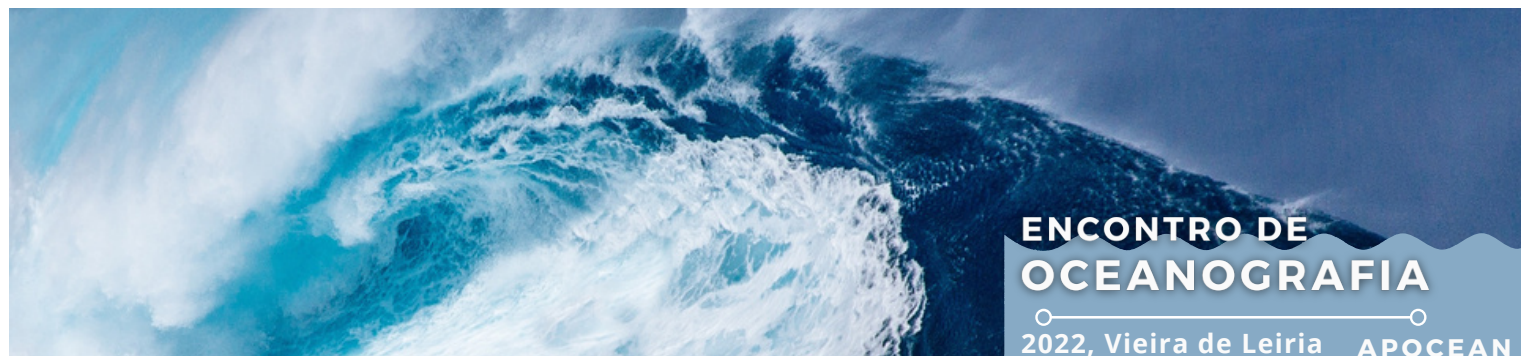
Figure 4. Ripples - creating pollution markers

Acknowledgements

This work was supported by "JUNO - Robotic exploration of Atlantic waters" project from the Fundação Luso-Americana para o Desenvolvimento, Project RAMP - Rapid-response to Marine Pollution, FA_06_2017_077 - funded by Direção Geral da Política do Mar (DGPM) through the Fundo Azul program and by GROOM-II funded by European Union's Horizon 2020 research and innovation programme under grant agreement No 951842.

Bibliography

- [1] K. B. Bennett and J. Flach. Ecological Interface Design: Thirty-Plus Years of Refinement, Progress, and Potential. *Human Factors*, June 2019.
- [2] A. S. Ferreira, J. Pinto, P. Dias, and J. B. de Sousa. The LSTS software toolchain for persistent maritime operations applied through vehicular ad-hoc networks. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, June 2017.
- [3] J. Pinto, P. Sousa Dias, and J. Borges de Sousa. Ripples: a tool for supervision and control of remote assets. *Instrumentation Viewpoint*, (20), 2018.
- [4] K. Vicente and J. Rasmussen. Ecological interface design: theoretical foundations. *IEEE Transactions on Systems, Man, and Cybernetics*, (4), July 1992.



CERTIFICADO DE PARTICIPAÇÃO

A **Associação Portuguesa de Oceanografia** certifica que

Afonso Sá

apresentou o trabalho "Ripples: Increasing human supervision over operational assets for a persistent and efficient ocean observation" no **Encontro de Oceanografia 2022**, que decorreu nos dias 6 e 7 de Junho em Vieira de Leiria.



Álvaro Peliz
Presidente da Associação Portuguesa de Oceanografia