



Gonçalo José Almeida Gomes

DESENVOLVIMENTO E IMPLEMENTAÇÃO DE UM SISTEMA DE MONITORIZAÇÃO DA QUALIDADE DA ÁGUA EM AMBIENTE MARINHO

Relatório de Estágio do Mestrado em Tecnologias de Informação Geográfica, área de especialização em Ciências e Tecnologias de Informação Geográfica, orientado pela Professora Doutora Cidália Maira Parreira da Costa Fonte e pelo Mestre Joaquim António Saraiva Patriarca e apresentado ao Departamento de Matemática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Janeiro de 2024



UNIVERSIDADE DE COIMBRA

Gonçalo José Almeida Gomes

DESENVOLVIMENTO E IMPLEMENTAÇÃO DE UM SISTEMA DE MONITORIZAÇÃO DA QUALIDADE DA ÁGUA EM AMBIENTE MARINHO

Relatório de Estágio do Mestrado em Tecnologias de Informação Geográfica, área de especialização em Ciências e Tecnologias de Informação Geográfica, orientado Pela Professora Doutora Cidália Maria Parreira da Costa Fonte e pelo Mestre Joaquim António Saraiva Patriarca e apresentado ao Departamento de Matemática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Janeiro de 2024



UNIVERSIDADE DE COIMBRA

Ficha Técnica

Tipo de trabalho	Relatório de Estágio
Título	Desenvolvimento e Implementação de um Sistema de Monitorização da Qualidade da Água em Ambiente Marinho
Autor/a	Gonçalo José Almeida Gomes
Orientador/a(s)	Doutora Cidália Maria Parreira da Costa Fonte Mestre Joaquim António Saraiva Patriarca
Júri	Presidente: Doutor João Manuel de Moraes Barros Fernandes
Identificação do Curso	Vogais: Doutor Alberto Jorge Lebre Cardoso Doutora Cidália Maria Parreira da Costa Fonte 2º Ciclo em Tecnologias de Informação Geográfica
Área científica	Tecnologias de Informação Geográfica
Especialidade/Ramo	Ciências e Tecnologias de Informação Geográfica
Ano	2024



UNIVERSIDADE DE COIMBRA

Agradecimentos

Em primeiro lugar, quero deixar um agradecimento aos meus orientadores, Professora Doutora Cidália Maria Parreira da Costa Fonte e Professor Joaquim António Saraiva Patriarca, por terem acreditado em mim e por ter tido a oportunidade de trabalhar com eles neste projeto. Obrigado por todo o conhecimento transmitido e pelo apoio ao longo desta etapa.

À minha família, em especial à minha irmã, pela paciência e amizade demonstrada ao longo desta etapa.

A todos os colegas com quem me cruzei durante este percurso académico, por o tornarem mais fácil e divertido.

Às minhas amigas mais próximas, pelo apoio, paciência e partilha de momentos únicos e memoráveis.

Por fim, agradeço a todo o corpo docente do mestrado em Tecnologias de Informação Geográfica e a todos com quem me cruzei durante este percurso e que não foram aqui mencionados.

A todos, muito obrigado!

Resumo

A água é um recurso fundamental à vida na terra, pelo que é necessário fazer a sua monitorização para servir de apoio a processos de tomada de decisão por parte de diversas entidades e em diversas atividades económicas. Uma dessas atividades é a aquicultura, na qual a qualidade da água assume um papel crucial, sendo necessário verificar se diversos parâmetros que definem a sua qualidade estão em conformidade com os valores de referência.

Com o objetivo de contribuir para a monitorização da qualidade da água e fornecer uma ferramenta capaz de garantir essa análise de forma contínua, foi realizado um estágio no Instituto de Engenharia de Sistemas e Computadores de Coimbra (INESC Coimbra). Neste âmbito foi desenvolvido o sistema gWater: um WebSIG que inclui ferramentas para monitorizar a qualidade da água em ambiente marinho, com recurso a dados do programa Copernicus.

O sistema foi desenvolvido com recurso a tecnologias de código aberto, tais como o Leaflet, GeoServer, GDAL, GRASS GIS, GeoPandas, Angular e Django. Estas tecnologias permitiram construir um WebSIG livre de custos associados à aquisição de licenças de *software*.

O resultado é um sistema que faz a recolha e tratamento de dados do CMEMS e Sentinel-2 de forma automática, e que permite aos utilizadores selecionar dados presentes num catálogo e adicioná-los à lista de camadas para visualização no mapa, com funcionalidades semelhantes às encontradas em qualquer *software* SIG ou WebSIG, como ativar e desativar a camada, *zoom to layer*, alteração da simbologia e preferências de visualização (opacidade e *thresholds*). Permite também executar ferramentas de processamento de dados geográficos e visualizar histogramas, gráficos de dispersão e gráficos de evolução temporal associados aos diversos tipos de dados. Os resultados obtidos através da execução das ferramentas são adicionados ao catálogo de dados, ficando disponíveis para consulta e visualização no mapa.

Palavras-chave: WebSIG; Monitorização da Qualidade da Água; Sentinel-2; CMEMS; GeoServer; Leaflet.

Abstract

Water is a fundamental resource for life on earth that requires monitoring to support decision making processes by various entities and economic activities. One of those activities is aquaculture, in which water quality plays a fundamental role to ensure that its standard parameters are in accordance with the reference values.

With the aim of contributing to water quality monitoring and providing a tool capable of ensuring an ongoing analysis, an internship was carried out at the Institute for Systems and Computers Engineering at Coimbra (INESCC). Within this context, the gWater system was developed: a WebGIS that includes tools for monitoring water quality in the marine environment, using data from the Copernicus program.

The system was developed using open-source technologies, such as Leaflet, GeoServer, GDAL, GRASS GIS, GeoPandas, Angular and Django. These technologies made it possible to build a WebGIS free of costs associated with software licenses.

The outcome is a system that automatically collects and processes CMEMS and Sentinel-2 data, allowing users to select data from a catalog and add it to the list of visualization layers on a map, with similar functionalities to the ones found in any GIS or WebGIS software, such as activating and deactivating layers, zoom to layer, changing symbology and display preferences (opacity and thresholds).

It also allows to run geographic data processing tools and visualize histograms, scatter plots and time evolution charts associated with different types of data. The results obtained with the execution of these tools are added to the data catalog, becoming available for consultation and visualization on the map.

Keywords: WebGIS; Water Quality Monitoring; Sentinel-2; CMEMS; GeoServer; Leaflet.

Índice Geral

AGRADECIMENTOS	II
RESUMO	III
ABSTRACT	IV
ÍNDICE GERAL	V
ÍNDICE DE FIGURAS	VII
ÍNDICE DE TABELAS	X
ABREVIATURAS, ACRÓNIMOS E SIGLAS	XIV
1. INTRODUÇÃO	1
1.1. ENQUADRAMENTO E MOTIVAÇÃO.....	1
1.2. OBJETIVOS	3
1.3. ESTRUTURA.....	4
2. DISPONIBILIZAÇÃO DE INFORMAÇÃO GEOGRÁFICA NA WEB: CONCEITOS FUNDAMENTAIS	6
2.1. SIG	6
2.2. SISTEMAS WEB	7
2.2.1. API e Web Service.....	10
2.3. WEBSIG.....	11
3. MONITORIZAÇÃO DA QUALIDADE DA ÁGUA	16
3.1. QUALIDADE DA ÁGUA	16
3.2. HARMFUL ALGAL BLOOM.....	16
3.3. MONITORIZAÇÃO COM DADOS OBTIDOS ATRAVÉS DE DETEÇÃO REMOTA.....	17
3.4. DISPOSITIVOS DE MONITORIZAÇÃO IN SITU	19
3.5. MONITORIZAÇÃO DA QUALIDADE DA ÁGUA E O PAPEL DOS WEBSIG.....	21
3.6. EXEMPLOS DE WEBSIG QUE REALIZAM A MONITORIZAÇÃO DA QUALIDADE DA ÁGUA.....	22
3.6.1. MyOcean	23
3.6.2. WorldView	24
3.6.3. COASTNET.....	24
3.7. ANÁLISE E COMPARAÇÃO DOS SISTEMAS.....	25
4. DRIVERS ARQUITETURAIS	27
4.1. REQUISITOS FUNCIONAIS	28
4.2. CONSTRANGIMENTOS TÉCNICOS	32
4.3. ATRIBUTOS DE QUALIDADE	34
5. ARQUITETURA DO SISTEMA	39
5.1. MODELO C4.....	39
5.1.1. Diagrama de Contexto	40

5.1.2.	Diagrama de Contentores.....	42
5.1.3.	Diagrama de Componentes.....	45
6.	TECNOLOGIAS UTILIZADAS.....	51
6.1.	FRONTEND.....	51
6.1.1.	DOM.....	51
6.1.2.	HTML.....	53
6.1.3.	CSS.....	53
6.1.4.	JavaScript.....	54
6.1.5.	TypeScript.....	54
6.1.6.	<i>Frameworks</i>	54
6.1.6.1.	Angular.....	54
6.1.6.2.	React.....	55
6.1.6.3.	Vue.js.....	55
6.1.6.4.	Análise e Comparação de <i>Frameworks</i>	56
6.1.7.	NgRx.....	57
6.2.	BACKEND.....	58
6.2.1.	Django e Django REST <i>Framework</i>	58
6.2.2.	GeoServer.....	59
6.3.	PROTOCOLO MQTT.....	60
7.	IMPLEMENTAÇÃO.....	61
7.1.	SCRIPTS DE RECOLHA E TRATAMENTO DOS DADOS.....	61
7.1.1.	Dados do CMEMS.....	62
7.1.2.	Sentinel-2.....	63
7.2.	DESENVOLVIMENTO API.....	66
7.2.1.	Aplicações Django.....	69
7.3.	APLICAÇÃO WEB.....	73
7.3.1.	Componentes.....	74
7.3.2.	Serviços.....	75
7.3.3.	Gestão do Estado da Aplicação.....	77
7.4.	EXECUÇÃO DE FERRAMENTAS.....	78
7.4.1.	Lógica de Funcionamento com Protocolo MQTT.....	78
7.4.2.	<i>Scripts</i> do Módulo gWater Toolbox.....	79
7.5.	BASE DE DADOS.....	82
7.6.	DISTRIBUIÇÃO DO TRABALHO.....	86
8.	RESULTADOS E DISCUSSÃO.....	89
9.	CONCLUSÕES.....	107
	BIBLIOGRAFIA.....	109
	APÊNDICES.....	112
	APÊNDICE A: DIAGRAMAS E TABELAS DE CASOS DE USO.....	113
	APÊNDICE B: ENDPOINTS.....	184

Índice de Figuras

Figura 1 - Componentes de um sistema web e as suas relações.....	8
Figura 2 - Esquema representativo do funcionamento geral do protocolo HTTP	9
Figura 3 - Xerox PARC Map Viewer.	12
Figura 4 - Johns Hopkins COVID-19 Dashboard, que serviu para monitorizar a evolução da pandemia COVID-19 em tempo real, tornando-se numa referência global para o estudo da propagação e evolução do coronavírus.....	13
Figura 5 - Partes integrantes de um WebSIG.....	15
Figura 6 - Descrição do processo de deteção remota e os elementos que o compõem.	18
Figura 7 - Espectro eletromagnético.	19
Figura 8 - Undersee device.	20
Figura 9 - Ilustração do funcionamento dos Undersee Devices em diferentes localizações.	21
Figura 10 - Página principal da plataforma MyOcean.....	23
Figura 11 - Página principal da plataforma WorldView.....	24
Figura 12 - Página principal da plataforma COASNET.	25
Figura 13 - Legenda dos diagramas de contexto, contentores e componentes.....	40
Figura 14 - Diagrama de Contexto do Sistema gWater.....	42
Figura 15 - Diagrama de Contentores do Sistema gWater.	43
Figura 16 - Diagrama de componentes do contentor "Aplicação web" do sistema gWater.....	46
Figura 17 - Diagrama de componentes do contentor "API" do sistema gWater.	49
Figura 18 - Exemplo de informação obtida após clicar no botão para inspecionar uma página em www.uc.pt	52
Figura 19 - Ciclo de funcionamento do NgRx.....	57
Figura 20 - Figura representativa dos passos/scripts de recolha e tratamento dos dados do CMEMS. 62	
Figura 21 - Esquema representativo do fluxo de processos do <i>script</i> responsável por calcular os intervalos necessários para a construção do estilo e da legenda.....	63
Figura 22 - Figura representativa dos passos/scripts de recolha e tratamento dos dados Sentinel-2.....	64
Figura 23 - Esquema representativo do fluxo de processos com vista à criação de estilos para as imagens do Sentinel-2.....	65
Figura 24 - Combinações de bandas para as imagens Sentinel-2.	65
Figura 25 - URLs utilizados para executar operações relacionadas com as layers no GeoServer.	71
Figura 26 - View correspondente ao método GET da classe responsável por lidar com uma <i>layer</i> específica do GeoServer.	71
Figura 27 - Esquema representativo da interface da página de visualização e manipulação dos dados. 74	
Figura 28 - Método <code>getHistogramData</code> do serviço <code>ToolschartsService</code>	76
Figura 29 - <i>Reducer</i> NgRx que altera a propriedade " <i>active</i> " de um objeto " <i>Layer</i> ".	77
Figura 30 - Esquema demonstrativo da lógica de comunicação entre a aplicação de cliente e os <i>scripts</i> de execução de ferramentas, através do protocolo MQTT.....	79
Figura 31 - Esquema representativo do fluxo de funcionamento dos <i>scripts</i> responsáveis pela execução de ferramentas no sistema gWater.....	79
Figura 32 - Diagrama de Entidade e Relacionamento de parte das tabelas criadas para assegurar a execução de ferramentas no sistema gWater.	80

Figura 33 - Modelo Django utilizado para a criação da tabela que guarda os dados associados às camadas CMEMS.	83
Figura 34 - Estrutura da Base de Dados Relacional do sistema gWater.	84
Figura 35 - Página de login do sistema gWater.	89
Figura 36 - Página principal do sistema gWater.	90
Figura 37 - Página principal com vários nós do catálogo de dados abertos.	91
Figura 38 - Página principal com duas camadas ativas.	92
Figura 39 - Página principal com um histograma.	93
Figura 40 - Histograma criado no sistema gWater através de dados CMEMS.	94
Figura 41 - Histograma criado no sistema gWater através de dados Sentinel-2.	94
Figura 42 - Página principal com um gráfico de dispersão.	94
Figura 43 - Menu que permite obter o gráfico de evolução temporal.	96
Figura 44 - Página principal com o catálogo das ferramentas aberto do lado direito da página.	97
Figura 45 - Página principal com uma ferramenta aberta.	98
Figura 46 - Exemplo de indicação do progresso da execução de uma ferramenta.	98
Figura 47 - Página principal com uma camada WFS relacionada com o aparecimento de algas, resultado da execução dos <i>scripts</i> que realizam este processamento de forma automática.	99
Figura 48 - Exemplo de camada WFS relacionada com a ocorrência de algas, resultado de uma execução de ferramenta.	100
Figura 49 - Opções disponíveis em cada camada do Map Layers.	102
Figura 50 - Janela que permite alterar a simbologia de uma camada CMEMS.	102
Figura 51 - Janela que permite alterar a simbologia de uma camada Sentinel-2.	102
Figura 52 - Janela que permite alterar a simbologia de uma camada relacionada com a ocorrência de algas.	103
Figura 53 - Janela das preferências de visualização de uma camada.	103

Apêndice A

Figura 54 - Diagrama de casos de uso relacionado com a autenticação no sistema gWater.	113
Figura 55 - Diagrama de casos de uso relacionado com a gestão dos utilizadores.	116
Figura 56 - Diagrama de casos de uso relacionado com o catálogo de dados do sistema gWater.	120
Figura 57 - Diagrama de casos de uso associado às operações do Map Layers.	122
Figura 58 - Diagrama de casos de uso relacionado com a apresentação de gráficos no sistema gWater.	127
Figura 59 - Diagrama de casos de uso relacionado com a execução de ferramentas no sistema gWater.	130
Figura 60 - Diagrama de casos de uso relacionado com a gestão dos estilos das Camadas CMEMS.	132
Figura 61 - Diagrama de casos de uso relacionados com a gestão de estilo no GeoServer.	136
Figura 62 - Diagrama de casos de uso relacionado com a gestão dos workspaces do GeoServer.	139
Figura 63 - Diagrama de casos de uso relacionado com a gestão das stores do GeoServer.	142
Figura 64 - Diagrama de casos de uso relacionado com a gestão das camadas do GeoServer.	145
Figura 65 - Diagrama de casos de uso relacionado com a gestão das variáveis do sistema gWater.	148
Figura 66 - Diagrama de casos de uso relacionado com a gestão das variáveis CMEMS.	152
Figura 67 - Diagrama de casos de uso relacionado com a gestão de mosaicos do Sentinel-2.	156
Figura 68 - Diagrama de casos de uso relacionado com a gestão dos <i>Algae Bloom</i> Datasets.	160
Figura 69 - Diagrama de casos de uso relacionado com a gestão de camadas <i>Algae Bloom</i>	164

Figura 70 - Diagrama de casos de uso relacionado com a gestão das camadas CMEMS do sistema gWater.....	168
Figura 71 - Diagrama de casos de uso relacionado com a gestão das camadas Sentinel-2 do sistema gWater.....	172
Figura 72 - Diagrama de casos de uso relacionado com a gestão das legendas das camadas CMEMS.	176
Figura 73 - Diagrama de casos de uso relacionado com a gestão das áreas de interesse do sistema gWater.....	180

Índice de Tabelas

Tabela 1 - Requisitos funcionais do sistema gWater.	29
Tabela 2 - Constrangimento técnico relacionado com a utilização do Angular.....	32
Tabela 3 - Constrangimento técnico relacionado com a aquisição dos dados do Sentinel-2.	32
Tabela 4 - Constrangimento técnico relacionado com a equipa de desenvolvimento.....	33
Tabela 5 - Constrangimento técnico relacionado com o hardware.....	33
Tabela 6 - Constrangimento técnico relacionado com o servidor.	33
Tabela 7 - Atributo de qualidade relacionado com a segurança do Sistema gWater.....	34
Tabela 8 - Atributo de qualidade relacionado com a interoperabilidade do Sistema gWater.	35
Tabela 9 - Atributo de qualidade relacionado com a precisão do Sistema gWater.	36
Tabela 10 - Atributo de qualidade relacionado com a usabilidade do sistema gWater.	37
Tabela 11 - Atributo de qualidade relacionado com a flexibilidade e adaptabilidade do Sistema gWater.	38
Tabela 12 - Critérios de comparação entre os diferentes <i>frameworks</i> de <i>frontend</i>	56
Tabela 13 - Códigos de status de respostas HTTP	68
Tabela 14 - Exemplo de códigos de status HTTP associados aos códigos específicos do sistema gWater.....	68

Apêndice A

Tabela 15 - Caso de uso "Login".....	113
Tabela 16 – Caso de uso "Alterar Palavra-passe".....	114
Tabela 17 – Caso de uso "Recuperar Palavra-passe".	114
Tabela 18 - Caso de uso "Logout".....	115
Tabela 19 – Caso de uso “Obter Utilizadores”	116
Tabela 20 – Caso de uso “Criar Utilizador”	117
Tabela 21 - Caso de uso “Eliminar Utilizadores”	117
Tabela 22 - Caso de uso “Obter Utilizador Específico”.	118
Tabela 23 - Caso de uso “Atualizar Utilizador”.	118
Tabela 24 - Caso de uso “Eliminar Utilizador Específico”.....	119
Tabela 25 - Caso de uso "Visualizar camadas disponíveis".....	120
Tabela 26 - Caso de uso "Adicionar Camada ao Map Layers".....	121
Tabela 27 - Caso de uso "Visualizar Camadas do Map Layers".....	122
Tabela 28 - Caso de uso "Remover Camada do Map Layers".	123
Tabela 29 - Caso de uso "Ativar Camada do Map Layers".	123
Tabela 30 - Caso de uso "Desativar Camada do Map Layers".....	124
Tabela 31 - Caso de uso "Visualizar Legenda".....	124
Tabela 32 - Caso de uso "Fazer Zoom para a Camada do Map Layers”.....	125
Tabela 33 - Caso de uso "Alterar Ordem das Camadas no Map Layers".	125
Tabela 34 - Caso de uso "Alterar a Simbologia das Camadas do CMEMS".....	126
Tabela 35 - Caso de uso "Alterar a Composição das Bandas das Cmadas Sentinel-2”.	126
Tabela 36 - Caso de uso "Visualizar Histograma associado a uma camada".	127
Tabela 37 - Caso de uso "Visualizar Gráfico de Dispersão".....	128

Tabela 38 - Caso de uso "Visualizar Gráfico de Evolução Temporal".....	129
Tabela 39 - Caso de uso "Visualizar Lista de Ferramentas".....	130
Tabela 40 - Caso de uso "Executar Ferramenta".....	131
Tabela 41 - Caso de uso "Visualizar Resultados Produzidos pelas Ferramentas".....	131
Tabela 42 - Caso de uso "Visualizar Estilos CMEMS".....	132
Tabela 43 - Caso de uso "Criar Estilo CMEMS".....	133
Tabela 44 - Caso de uso "Eliminar Todos os Estilos das Camadas CMEMS".....	133
Tabela 45 - Caso de uso "Obter Estilo Específico das Camadas CMEMS".....	134
Tabela 46 - Caso de uso "Editar Estilo das Camadas CMEMS".....	134
Tabela 47 - Caso de uso "Remover Estilo das Camadas CMEMS".....	135
Tabela 48 - Caso de uso "Obter Estilos do GeoServer".....	136
Tabela 49 - Caso de uso "Criar Estilo do GeoServer".....	137
Tabela 50 - Caso de uso "Editar Estilo do GeoServer".....	138
Tabela 51 - Caso de uso "Eliminar Estilo do GeoServer".....	138
Tabela 52 - Caso de uso "Obter Workspaces".....	139
Tabela 53 - Caso de uso "Criar <i>Workspace</i> ".....	140
Tabela 54 - Caso de uso "Obter <i>Workspace</i> Específico".....	140
Tabela 55 - Caso de uso "Eliminar <i>Workspace</i> ".....	141
Tabela 56 - Caso de uso "Obter <i>Stores</i> ".....	142
Tabela 57 - Caso de uso "Criar <i>Store</i> ".....	143
Tabela 58 - Caso de uso "Obter <i>Store</i> Específica".....	143
Tabela 59 - Caso de uso "Eliminar <i>Store</i> ".....	144
Tabela 60 - Caso de uso "Obter Camadas do GeoServer".....	145
Tabela 61 - Caso de uso "Criar Camada do GeoServer".....	146
Tabela 62 - Caso de uso "Obter Camada Específica do GeoServer".....	146
Tabela 63 - Caso de uso "Atualizar Camada do GeoServer".....	147
Tabela 64 - Caso de uso "Eliminar Camada do GeoServer".....	147
Tabela 65 - Caso de uso "Obter Variáveis".....	148
Tabela 66 - Caso de uso "Criar Variável".....	149
Tabela 67 - Caso de uso "Eliminar Variáveis".....	149
Tabela 68 - Caso de uso "Obter Variável Específica".....	150
Tabela 69 - Caso de uso "Editar Variável".....	150
Tabela 70 - Caso de uso "Eliminar Variável Específica".....	151
Tabela 71 - Caso de uso "Obter Variáveis CMEMS".....	152
Tabela 72 - Caso de uso "Criar Variável CMEMS".....	153
Tabela 73 - Caso de uso "Eliminar Variáveis CMEMS".....	153
Tabela 74 - Caso de uso "Obter Variável CMEMS Específica".....	154
Tabela 75 - Caso de uso "Editar Variável CMEMS".....	154
Tabela 76 - Caso de uso "Eliminar Variável CMEMS Específica".....	155
Tabela 77 - Caso de uso "Obter Mosaicos do Sentinel-2".....	156
Tabela 78 - Caso de uso "Criar Mosaico do Sentinel-2".....	157
Tabela 79 - Caso de uso "Eliminar Mosaicos do Sentinel-2".....	157
Tabela 80 - Caso de uso "Obter Mosaico do Sentinel-2 Específico".....	158
Tabela 81 - Caso de uso "Editar Mosaico do Sentinel-2".....	158

Tabela 82 - Caso de uso "Eliminar Mosaico do Sentinel-2 Específico".	159
Tabela 83 - Caso de uso "Obter Datasets <i>Algae Bloom</i> ".	160
Tabela 84 - Caso de uso "Criar um <i>Algae Bloom</i> Dataset".	161
Tabela 85 - Caso de uso "Eliminar <i>Algae Bloom</i> Datasets".	161
Tabela 86 - Caso de uso "Obter um <i>Algae Bloom</i> Dataset Específico".	162
Tabela 87 - Caso de uso "Editar um <i>Algae Bloom</i> Dataset".	162
Tabela 88 - Caso de uso "Eliminar um <i>Algae Bloom</i> Dataset Específico".	163
Tabela 89 - Caso de uso "Obter Camadas <i>Algae Bloom</i> ".	164
Tabela 90 - Caso de uso "Criar Camada <i>Algae Bloom</i> ".	165
Tabela 91 - Caso de uso "Eliminar Camadas <i>Algae Bloom</i> ".	165
Tabela 92 - Caso de uso "Obter Camada <i>Algae Bloom</i> Específica".	166
Tabela 93 - Caso de uso "Editar Camada <i>Algae Bloom</i> ".	166
Tabela 94 - Caso de uso "Eliminar Camada <i>Algae Bloom</i> Específica".	167
Tabela 95 - Caso de uso "Obter Camadas CMEMS".	168
Tabela 96 - Caso de uso "Criar Camada CMEMS".	169
Tabela 97 - Caso de uso "Eliminar Camadas CMEMS".	169
Tabela 98 - Caso de uso "Obter Camada CMEMS Específica".	170
Tabela 99 - Caso de uso "Editar Camada CMEMS".	170
Tabela 100 - Caso de uso "Eliminar Camada CMEMS Específica".	171
Tabela 101 - Caso de uso "Obter Camadas Sentinel-2".	172
Tabela 102 - Caso de uso "Criar Camada Sentinel-2".	173
Tabela 103 - Caso de uso "Eliminar Camadas Sentinel-2".	173
Tabela 104 - Caso de uso "Obter Camada Sentinel-2 Específica".	174
Tabela 105 - Caso de uso "Editar Camada Sentinel-2".	174
Tabela 106 - Caso de uso "Eliminar Camada Sentinel-2 Específica".	175
Tabela 107 - Caso de uso "Obter Legendas CMEMS".	176
Tabela 108 - Caso de uso "Criar Legenda CMEMS".	177
Tabela 109 - Caso de uso "Eliminar Legendas CMEMS".	177
Tabela 110 - Caso de uso "Obter Legenda CMEMS Específica".	178
Tabela 111 - Caso de uso "Editar Legenda CMEMS".	178
Tabela 112 - Caso de uso "Eliminar Legenda CMEMS Específica".	179
Tabela 113 - Caso de uso "Obter Áreas de Interesse".	180
Tabela 114 - Caso de uso "Criar Área de Interesse".	181
Tabela 115 - Caso de uso "Eliminar Áreas de Interesse".	181
Tabela 116 - Caso de uso "Obter Área de Interesse Específica".	182
Tabela 117 - Caso de uso "Editar Área de Interesse".	182
Tabela 118 - Caso de uso "Eliminar Área de Interesse Específica".	183

Apêndice B

Tabela 119 - Endpoints relacionados com a autenticação no sistema gWater.	184
Tabela 120 - Endpoints relacionados com a gestão dos utilizadores.	184
Tabela 121 - Endpoints relacionados com os pedidos ao GeoServer.	184
Tabela 122 - Endpoints relacionados com os gráficos.	185
Tabela 123 - Endpoints relacionados com as ferramentas.	185

Tabela 124 - Endpoints relacionados com os parâmetros das ferramentas.....	186
Tabela 125 - Endpoints relacionados com a execução de ferramentas.....	186
Tabela 126 - Endpoints relacionados o resultado da execução de ferramentas.....	187
Tabela 127 - Outros endpoints relacionados com a execução de ferramentas.....	187
Tabela 128 - Endpoints relacionados co os estilos das camadas CMEMS.....	188
Tabela 129 - Endpoints relacionados com os estilos do GeoServer.....	188
Tabela 130 - Endpoints relacionados com os workspaces do GeoServer.....	189
Tabela 131 - Endpoints relacionados com as stores do GeoServer.....	189
Tabela 132 - Endpoints relacionados com as camadas do GeoServer.....	189
Tabela 133 - Endpoints relacionados com as variáveis.....	190
Tabela 134 - Endpoints relacionados com as variáveis do CMEMS.....	190
Tabela 135 - Endpoints relacionados com os mosaicos do Sentinel-2.....	191
Tabela 136 - Endpoints relacionados com os <i>Algae Bloom</i> Datasets.....	191
Tabela 137 - Endpoints relacionados com as camadas <i>Algae Bloom</i>	192
Tabela 138 - Endpoints relacionados com as camadas CMEMS.....	192
Tabela 139 - Endpoints relacionados com as camadas Sentinel-2.....	193
Tabela 140 - Endpoints relacionados com as legendas das camadas CMEMS.....	193
Tabela 141 - Endpoints relacionados com as áreas de interesse.....	194
Tabela 142 - Outros <i>endpoints</i>	194

Abreviaturas, Acrónimos e Siglas

API	<i>Application Programming Interface</i>
CERN	<i>Conseil Européen pour la Recherche Nucléaire</i>
CMEMS	<i>Copernicus Marine Environment Monitoring Service</i>
CSS	<i>Cascading Style Sheets</i>
CSV	<i>Comma-Separated Values</i>
DOM	<i>Document Object Model</i>
ESA	<i>European Space Agency</i>
FAO	<i>Food and Agriculture Organization</i>
GDAL	<i>Geospatial Data Abstraction Library</i>
GRASS	<i>Geographic Resources Analysis Support System</i>
HAB	<i>Harmful Algal Bloom</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IDE	<i>Integrated Development Environment</i>
IETF	<i>Internet Engineering Task Force</i>
INESC	Instituto de Engenharia de Sistemas e Computadores
IoT	<i>Internet of Things</i>
JPEG	<i>Joint Photographic Experts Group</i>
JSON	<i>JavaScript Object Notation</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
NASA	<i>National Aeronautics and Space Administration</i>
OGC	<i>Open Geospatial Consortium</i>
ORM	<i>Object Relational Mapping</i>
PNG	<i>Portable Network Graphics</i>
REST	<i>REpresentational State Transfer</i>

SIG	Sistema de Informação Geográfica
SLD	Styled Layer Descriptor
SOAP	<i>Simple Object Access Protocol</i>
TIF	<i>Tagged Image File Format</i>
UML	<i>Unified Modeling Language</i>
URL	<i>Uniform Resource Locator</i>
VANT	Veículo Aéreo Não Tripulado
W3C	<i>World Wide Web Consortium</i>
WFS	<i>Web Feature Service</i>
WMS	<i>Web Map Service</i>
WWW	<i>World Wide Web</i>

I. Introdução

I.1. Enquadramento e Motivação

O presente relatório enquadra-se na unidade curricular “Relatório de Estágio em Ciências de Informação Geográfica” do Mestrado em Tecnologias de Informação Geográfica, ramo de Ciências e Tecnologias de Informação Geográfica, coordenado pela Faculdade de Ciências e Tecnologias da Universidade de Coimbra, no ano letivo de 2023/2024.

O trabalho desenvolvido enquadra-se também numa bolsa BII (Bolsa de Iniciação à Investigação) do Instituto de Engenharia de Sistemas e Computadores de Coimbra (INESC Coimbra), tendo como área de aplicação o desenvolvimento de um WebSIG capaz de realizar a monitorização da qualidade da água. O mestrando desempenhou o papel de desenvolvedor, numa equipa que contou com outros dois elementos, um responsável pelo desenvolvimento e um líder de equipa.

Os WebSIG têm vindo a desempenhar um papel fundamental na partilha e disseminação de informação com carácter geoespacial, permitindo a interação das pessoas com sistemas de informação geográfica de forma instantânea e em qualquer lugar, atuando como uma ferramenta valiosa na tomada de decisões. Assim, os WebSIG transformaram a forma como partilhamos informação geográfica e forneceram as ferramentas necessárias para a monitorização, análise, planeamento e apoio à decisão, servindo governos, negócios, ciência e cidadãos comuns.

Entre os setores da indústria que beneficiam das importantes informações provenientes dos WebSIG, destaca-se o setor alimentar, nomeadamente as atividades ligadas à aquicultura, onde a qualidade da água assume um papel crucial.

Como forma de dar resposta às necessidades do setor alimentar, a indústria da aquicultura tem apresentado um crescimento significativo, tornando-se assim num setor estratégico. De acordo com os dados mais recentes das Estatísticas da Pesca da Direção-Geral de Recursos Naturais, Segurança e Serviços Marítimos, existiam em Portugal no final de 2021 um total de 1 252 estabelecimentos licenciados em aquicultura, sendo que 1 242 correspondem a águas de transição¹ e marinhas, ocupando 2 957 hectares².

¹ Águas de transição são massas de água de superfície junto à foz dos rios que se apresentam como sendo uma mistura de água doce com água salgada, devido à sua proximidade com águas costeiras, sendo normalmente designadas como águas salobras.

² Publicações Estatísticas da Pesca – INE/DGRM. <https://www.dgrm.mm.gov.pt/esta> (acedido em 17 de julho, 2023)

Contudo, e apesar do crescimento deste setor da indústria, é necessário dar resposta a alguns desafios que se colocam. Um desses desafios é a perda de produção, que está associado à alteração do estado da água, podendo resultar numa menor eficiência e levantar questões de sustentabilidade de produção. Estas alterações no estado da água estão associadas a oscilações nos valores das variáveis que são utilizadas para medir a sua qualidade, tais como a temperatura, clorofila ou salinidade, sendo estas oscilações a principal causa desta fragilidade, uma vez que desempenham um papel fundamental para a sobrevivência e para o crescimento saudável das produções (Boyd, 2015).

No contexto acima descrito, em que a água é um elemento essencial, é necessário haver uma monitorização e controlo do seu estado, de maneira a garantir o equilíbrio desejado. No entanto, esse estado depende da finalidade das atividades. No caso da aquicultura, o tipo de produção e o tipo de pescado produzido obrigam a diferentes estados ideais de qualidade da água, sendo necessário definir as variáveis consideradas críticas e indicando para cada espécie um valor referência e um intervalo de valores seguros (Li & Liu, 2018).

Além disso, esta indústria lida com produtos alimentares, tendo o dever de seguir as recomendações indicadas nos relatórios de segurança e qualidade da água utilizada na produção e processamento de peixe e produtos da pesca da Organização das Nações Unidas para a Alimentação e a Agricultura (FAO) ³.

Uma vez definidas as variáveis consideradas críticas e os respetivos valores aceitáveis, é necessário garantir uma constante monitorização das mesmas, possibilitando uma rápida atuação por parte dos produtores caso ocorram oscilações relevantes, de forma a evitar perdas ou diminuição da eficiência das produções. Surge então a necessidade de obtenção e tratamento de dados de forma constante com o objetivo de fornecer aos produtores os meios necessários para efetuar a respetiva monitorização. É com base neste ponto de partida que surge o projeto gWater apresentado neste relatório, um WebSIG que fornece ferramentas capazes de realizar a monitorização contínua da qualidade da água em ambiente marinho através de dados produzidos pelo programa Copernicus⁴, permitindo a tomada de ações por parte dos utilizadores de forma atempada sempre que se verifiquem oscilações na sua qualidade.

De notar que, apesar deste projeto incidir sobre esta temática, o sistema desenvolvido pode ser aplicado noutros contextos, tais como a monitorização da poluição atmosférica ou do uso do

³ Relatório disponível em https://www.google.pt/books/edition/_/gOnXEAAAQBAJ?hl=pt-PT&gbpv=1&pg=PP1

⁴ O programa Copernicus é um programa de observação da terra liderado pela Comissão Europeia, com o apoio da Agência Espacial Europeia (ESA).

solo, seja através de novos dados e respetivas fontes ou de dados já existentes neste projeto, como é o caso das imagens da constelação de satélites Sentinel-2, que podem ter diversas aplicações. Além dos satélites Sentinel-2, existem outras séries de satélites do programa Copernicus, que é um projeto da agência espacial europeia e da comissão europeia que disponibiliza dados de forma gratuita, podendo servir de base para diversas aplicações, tal como as referidas acima.

1.2. Objetivos

O objetivo geral deste estágio é o desenvolvimento um sistema de monitorização da qualidade da água, que será disponibilizado aos utilizadores através de uma aplicação web. Este sistema terá como principal função fornecer os dados e informação necessária para uma monitorização e controlo da qualidade da água adequada nos locais de interesse associados a cada utilizador.

Entre as principais funcionalidades do sistema destacam-se a recolha sistemática e pré-processamento dos dados com a criação automática da simbologia necessária, visualização dos dados geográficos num mapa, o que implica a utilização de um servidor de mapas e a criação de serviços, e visualização de representações visuais dos dados através de gráficos com linhas, barras ou pontos, tais como gráficos de evolução temporal, histogramas e gráficos de dispersão, de forma a facilitar a compreensão de tendências, padrões e relações presentes nos mesmos. Além disso, essa página deve contar ainda com a possibilidade de execução de ferramentas de processamento de dados geográficos, permitindo manipular os dados do *Copernicus Marine Environment Monitoring Service* (CMEMS) e do Sentinel-2 de forma a obter informações úteis que possam contribuir para a tomada de decisões acertadas por parte dos utilizadores.

Adicionalmente devem ser elaborados os requisitos funcionais associados ao sistema gWater, assim como diagramas e tabelas de caso de uso que permitam entender toda a lógica de funcionamento do sistema, com particular foco na página de visualização de dados e execução de ferramentas.

As tecnologias utilizadas neste projeto são de código aberto, permitindo assim a construção de um WebSIG livre de custos associados à aquisição de licenças de *software*.

1.3. Estrutura

O presente relatório está dividido em nove capítulos, como se indica de seguida:

- O primeiro capítulo, **Introdução**, apresenta o enquadramento do presente relatório, a motivação, apresentando o contexto no qual se insere, os objetivos e a estrutura.
- O segundo capítulo, **Disponibilização de Informação Geográfica na Web: Conceitos Fundamentais**, pretende enquadrar o leitor acerca de temas relevantes para o entendimento do trabalho desenvolvido, através da definição de conceitos como Sistema de Informação Geográfica (SIG), Sistemas Web e WebSIG, servindo de base para o entendimento dos restantes capítulos.
- O terceiro capítulo, **Monitorização da Qualidade da Água**, enquadra o leitor acerca da área de aplicação do trabalho desenvolvido e apresentado neste relatório, apresentando os conceitos e os meios necessários para fazer a monitorização da qualidade da água. Neste capítulo são ainda apresentados diversos exemplos de WebSIG que têm como objetivo fazer essa monitorização, e ainda uma comparação entre os mesmos.
- No quarto capítulo, **Drivers Arquiteturais**, são apresentados os requisitos funcionais, os constrangimentos técnicos e ainda os atributos de qualidade.
- O quinto capítulo, **Arquitetura do Sistema**, apresenta a arquitetura do sistema através de diagramas baseados no modelo C4. Os diagramas encontram-se divididos em três níveis de abstração, o diagrama de contexto, o diagrama de contentores e o diagrama de componentes.
- O sexto capítulo, **Tecnologias Utilizadas**, apresenta as tecnologias utilizadas ao longo do processo de implementação, nomeadamente linguagens de programação e *frameworks* utilizados no *frontend* e *backend* do sistema gWater. Além disso, é também apresentado o protocolo MQTT, uma vez que é relevante no contexto da execução de ferramentas no sistema. Neste capítulo, foram excluídas tecnologias mais específicas que foram utilizadas na implementação deste sistema, nomeadamente bibliotecas Python como o GDAL ou GRASS que servem para lidar com dados geoespaciais. Foram também deixadas de fora tecnologias como a biblioteca JavaScript D3.js, que foi utilizada na construção dos gráficos na aplicação de cliente.

- No sétimo capítulo, **Implementação**, são apresentadas todas as etapas da implementação das funcionalidades previstas neste estágio. Este capítulo encontra-se dividido em várias secções. Em primeiro lugar, é detalhado o processo de implementação dos *scripts* de recolha e tratamento dos dados. Depois apresenta-se o desenvolvimento da API, abordando as diferentes partes do projeto de *backend* e apresentando diversos exemplos de funcionamento do sistema. De seguida é descrita a implementação da aplicação cliente. Na secção seguinte é detalhado o processo de implementação das funcionalidades que permitem a execução de ferramentas no sistema. Depois é apresentada a estrutura da base de dados relacional do sistema gWater, sendo detalhadas algumas das relações existentes entre tabelas e apresentada a sua justificação. Por último, é apresentada a distribuição do trabalho ao longo de todo o processo de implementação, de forma a clarificar as contribuições do mestrando.
- O oitavo capítulo, **Resultados e Discussão**, apresenta os resultados obtidos após o processo de implementação de parte das funcionalidades desejadas no sistema gWater, assim como uma discussão/análise desses resultados.
- Por último, no nono capítulo, **Conclusões**, apresentam-se as conclusões finais do autor e uma reflexão acerca dos desafios e dificuldades encontradas ao longo desta etapa, e dos resultados obtidos. Este capítulo apresenta ainda observações e sugestões acerca do trabalho que pode vir a ser desenvolvido neste sistema.

2. Disponibilização de Informação Geográfica na Web: Conceitos Fundamentais

Este capítulo apresenta e explica de forma sucinta alguns dos conceitos mais relevantes no contexto do projeto gWater, nomeadamente SIG e sistemas *web*, conceitos fundamentais para o entendimento geral do que é um WebSIG e como este funciona.

2.1. SIG

Um Sistema de Informação é um conjunto de componentes interrelacionadas que fazem a coleção, armazenamento, processamento e distribuição de dados. Esses dados, quando organizados e processados criam um valor adicional, gerando informação que irá dar origem a conhecimento capaz de auxiliar na tomada de decisões. Os dados contêm uma descrição elementar de transações, atividades, eventos ou outras ocorrências, apresentando valores não organizados e desprovidos de significado, enquanto a informação diz respeito a dados que foram organizados e processados de maneira a possuírem um significado e contexto. Por último, o conhecimento, enquanto resultado de um processo de interpretação da informação e do seu entendimento, contribui para gerar experiência e aprendizagem acumulada (Goodchild et al., 2005; Rainer & Prince, 2022).

“Quase tudo o que acontece, acontece algures. Em grande parte, nós, humanos, estamos confinados nas nossas atividades à superfície da Terra. (...) Por conseguinte, a localização geográfica é um atributo importante das atividades, políticas, estratégias e planos. Os sistemas de informação geográfica são uma classe especial de sistemas de informação que registam não apenas eventos, atividades e coisas, mas também onde esses eventos, atividades e coisas acontecem ou existem.” (Goodchild et al., 2005, p.4)

Um sistema de informação, é, portanto, um sistema que disponibiliza informação no momento certo, para as pessoas certas, na quantidade certa e no formato certo (Rainer & Prince, 2022). Estes sistemas, em contexto geográfico e quando baseados em computador, funcionam como uma associação de *hardware*, *software*, bases de dados, procedimentos e pessoas, que em conjunto fazem a recolha, armazenamento, manipulação, processamento, partilha e apresentação dos dados georreferenciados. Segundo Longley & Cheshire (2017), os SIG são sistemas utilizados para guardar e processar informação geográfica, podendo armazenar grandes quantidades de informação geográfica em bases de dados e realizar operações de forma significativamente mais eficiente relativamente à realização dessas mesmas tarefas de forma manual, automatizando assim

o processo de criação de mapas, que podem ser muito úteis em diversas situações e contribuir para uma maior rapidez de resposta em variados contextos.

2.2. Sistemas Web

Para compreender o que é um WebSIG, convém primeiro entender o que é um sistema *web* e como funciona.

*“The World Wide Web is like an encyclopaedia, a telephone directory, a record collection (...) all rolled into one and accessible through any computer”*⁵ (Gillies & Cailliau, 2000, p.1).

A *World Wide Web* é muitas vezes confundida com a internet, apesar de serem duas coisas diferentes. A internet é a infraestrutura que permite a existência da *World Wide Web*, tal como as estradas são as infraestruturas utilizadas por diferentes tipos de carros. A *World Wide Web* é como um desses carros que utiliza a infraestrutura existente.

A arquitetura da *World Wide Web* foi proposta por Tim Berners-Lee. Ele criou o primeiro servidor *web*, navegador de internet e página *web* no seu computador no CERN (Conseil Européen pour la Recherche Nucléaire - European Council for Nuclear Research)⁶.

Um sistema *web* encontram-se dividido em diversas componentes (ver Figura 1). Essas componentes podem ser agrupadas em camadas individuais. A camada de apresentação é a camada que permite aos utilizadores interagir com o sistema, através da interface de utilizador, e na Figura 1 está representada pelos dois clientes. A camada de aplicação contém a lógica do sistema, estando representada na Figura 1 pelo servidor *web*. Esta camada é responsável por fazer o processamento dos dados necessários para o funcionamento da aplicação *web*. Serve também de intermediária entre a camada de apresentação e a camada de dados, que é a camada responsável pelo armazenamento dos dados e pelo fornecimento dos mesmos sempre que são requisitados.

⁵ Este excerto pode traduzir-se para “A *World Wide Web* é como uma enciclopédia, uma lista telefónica, uma coleção de discos (...) tudo junto e acessível através de qualquer computador.”

⁶ <https://home.cern/science/computing/birth-web/short-history-web>

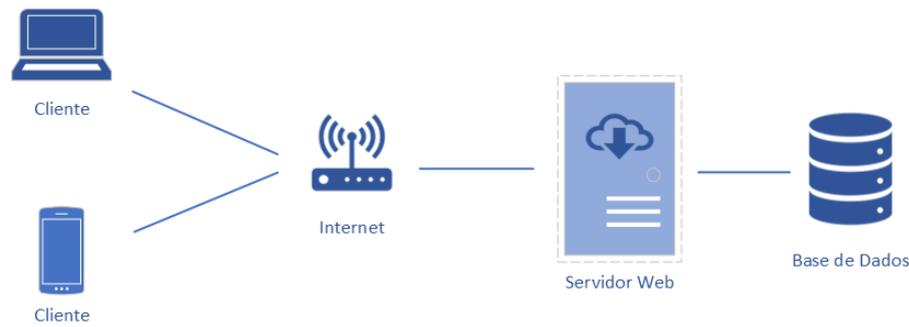


Figura 1 - Componentes de um sistema web e as suas relações.

O servidor *web* é responsável por aceitar e receber as solicitações feitas do lado cliente, processar a solicitação recebida, aceder a um recurso específico, gerar uma resposta e enviá-la (Moreira, 2022). Esta comunicação é feita normalmente utilizando o protocolo HTTP (Hypertext Transfer Protocol) criado por Tim Berners-Lee com o intuito inicial de partilha de documentos hipertexto⁷.

Os recursos existentes no servidor *web* têm de ser identificados, para que um cliente consiga localizar um recurso específico do servidor. Esta localização é feita através do *Uniform Resource Locator* (URL), que descreve a localização específica de um recurso existente num servidor. O URL especifica o protocolo, o servidor e o recurso específico.

Sempre que um utilizador introduz um URL válido no seu browser, que indica a localização específica de um determinado recurso, um servidor *web* recebe o seu pedido e dá início ao processo que levará informação de volta ao utilizador. Na Figura 2 está representado o fluxo de funcionamento do protocolo HTTP.

⁷ Hipertexto é uma forma de escrita e leitura eletrónica não linear através de ligações a outras partes do texto ou até outros documentos. Um exemplo deste princípio pode ser encontrado neste documento ao clicar numa referência a uma figura que contém uma ligação para essa figura.

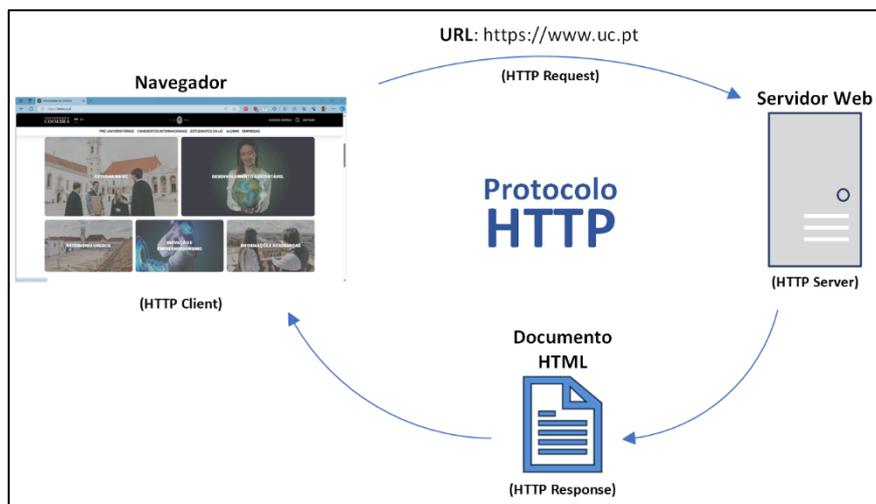


Figura 2 - Esquema representativo do funcionamento geral do protocolo HTTP

Neste caso o servidor devolve um documento HTML com o código fonte da página, que será interpretado pelo navegador. No entanto, estas respostas podem conter documentos noutros formatos, apesar de o HTML ser o mais utilizado. Existem também outros protocolos utilizados em cenários comuns na *web*, como, por exemplo, o *Message Queuing Telemetry Transport* (MQTT) ou o protocolo WebSocket, porém o HTTP é o mais utilizado para a comunicação entre cliente e servidor *web* aqui abordada (Gourley et al., 2002; Pollard, 2019).

Os pedidos feitos pelo cliente podem ser de diferentes tipos. Pode solicitar informação, pode submeter um conjunto de dados que ficará registado numa base de dados (ver Figura 1), pode alterar esses dados ou ainda eliminá-los. Para isso, deve ser especificado um dos seguintes métodos no pedido realizado pelo cliente (HTTP Request):

- GET – O método GET é utilizado para solicitar dados de um recurso específico. Os parâmetros do pedido são enviados no URL do pedido GET.
- POST – O método POST é utilizado para submeter dados. Estes dados são enviados para o servidor e normalmente estão associados a um novo registo numa tabela da base de dados. Os dados enviados através de um pedido HTTP com o método POST são guardados no corpo do pedido.
- PUT – O método PUT é normalmente utilizado para atualizar um registo específico existente numa base de dados.

- DELETE – O método DELETE remove um determinado recurso (pode eliminar um ou todos os registros de uma determinada tabela na base de dados).

De notar que esses pedidos não são feitos diretamente à base de dados. Os pedidos são feitos ao servidor *web* e este fica responsável por lidar com estes pedidos de diferentes tipos, respeitando a convenção padrão do funcionamento do protocolo HTTP. Para lidar com estes pedidos, deve existir uma *Application Programming Interface* (API) implementada no servidor *web*, de maneira a lidar corretamente e de maneira eficiente e estruturada com estes pedidos.

2.2.1. API e Web Service

Uma API é uma interface que permite a comunicação entre dois programas, especificando como deve ser feita essa comunicação. Desta forma, um programa de *software* pode interagir com uma aplicação existente. Por outras palavras, uma API é uma interface que permite o acesso de terceiros a dados ou recursos de uma aplicação.

Existe um tipo específico de API chamado de *Web Service*. Um *Web Service*, ao contrário de um *Website*, não é uma aplicação *web* que pode ser acedida pelos utilizadores, mas sim um conjunto de métodos que podem ser utilizados por outros programas na internet. É utilizado para transferir dados entre diferentes plataformas. Os *Web Services* permitem utilizar sistemas já existentes, integrando as suas funcionalidades no nosso sistema (Petrie, 2016).

De acordo com a *World Wide Web Consortium* (W3C), “*Web services provide a standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks.*” “*Programs providing simple services can interact with each other in order to deliver sophisticated added-value services*”⁸.

Para criar uma API, é necessário seguir uma arquitetura que permita definir as regras de comunicação com a mesma. As mais utilizadas são as arquiteturas REST (REpresentational State Transfer) cujo formato de partilha de dados mais utilizado é JSON (JavaScript Object Notation) e SOAP (Simple Object Access Protocol), que utiliza XML (Extensible Markup Language) como

⁸ Estes dois excertos retirados de <https://www.w3.org/wiki/WebServices> podem traduzir-se para “Os serviços *web* constituem um meio normalizado com vista à interoperabilidade entre diferentes aplicações de *software*, executadas numa variedade de plataformas e/ou *frameworks*.” “Os programas que fornecem serviços simples podem interagir entre si para fornecer serviços sofisticados de valor acrescentado”.

formato dos dados. No contexto dos *Web Services*, é mais comum a utilização da arquitetura SOAP (Richardson & Ruby, 2008).

No projeto gWater, a API construída com recurso à biblioteca Django REST Framework (Python) é baseada na arquitetura REST.

2.3. WebSIG

Definidos os conceitos de SIG e sistemas *web*, será apresentado de seguida o conceito de WebSIG, que é a combinação da *web* com os sistemas de informação geográfica.

Os WebSIG utilizam tecnologias *web* de maneira a permitir a interação das pessoas com sistemas de informação geográfica de forma instantânea e em qualquer lugar. Desde o desenvolvimento do primeiro SIG informatizado nos anos 60 por Roger Tomlinson, designado por “*Canada Geographic Information System*”, que estes têm vindo a evoluir de forma contínua, tendo sido no início dos anos 90 com o surgimento da *World Wide Web* que se deram os avanços que levariam ao surgimento dos WebSIG. Foi no ano de 1993 que surgiu um *software* que permitiu aos utilizadores utilizar um SIG sem tê-lo instalado localmente na sua máquina, chamado Xerox PARC Map Viewer (ver Figura 33) e criado no Palo Alto Research Center (Fu, 2018). De notar que é um pouco difícil determinar qual o primeiro WebSIG, uma vez que pode variar consoante as diferentes definições que se dá ao termo WebSIG.

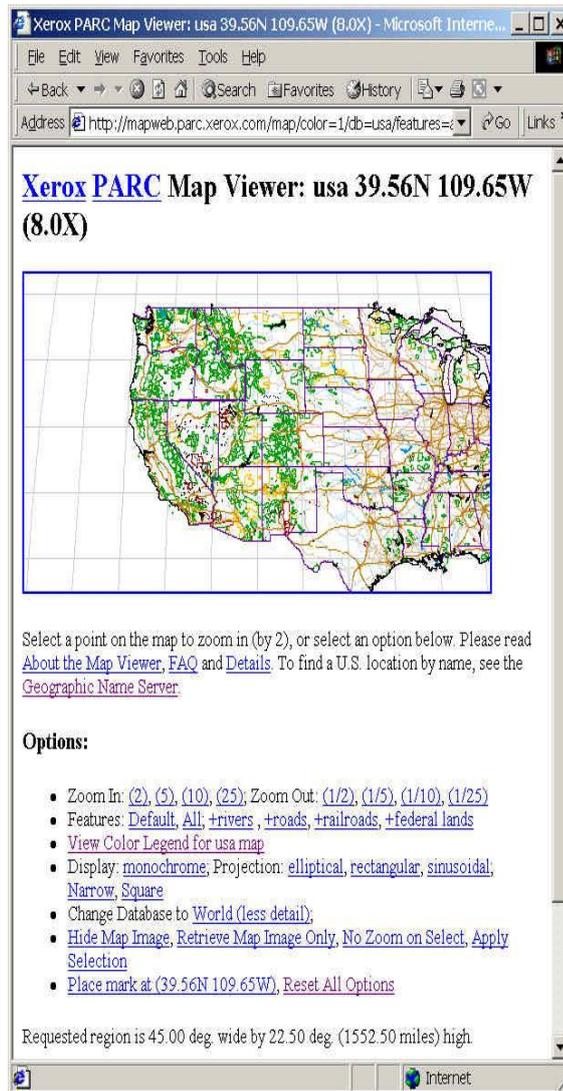


Figura 3 - Xerox PARC Map Viewer.

Fonte: <https://map.sdsu.edu/geog583/images/xerox-2.jpg>.

Entre as diversas vantagens de um WebSIG, destacam-se a facilidade de partilha de informação geográfica com organizações ou pessoas de todo o mundo, eliminação do risco de incompatibilidade entre sistemas operativos, uma vez que estes sistemas correm em navegadores da *web* (sendo que todos utilizam as mesmas tecnologias, tais como JavaScript) e são fáceis de utilizar, visto que os WebSIG são por norma intuitivos e não requerem experiência técnica de utilização de outros *softwares*. Além disso, todos os utilizadores podem beneficiar de atualizações e novas funcionalidades simultaneamente.

Os WebSIG transformaram a forma como partilhamos informação geográfica e forneceram as ferramentas necessárias para a monitorização, análise, planeamento e apoio à

decisão, servindo de apoio a governos, negócios, ciência e à vida quotidiana (ver exemplo na Figura 4).

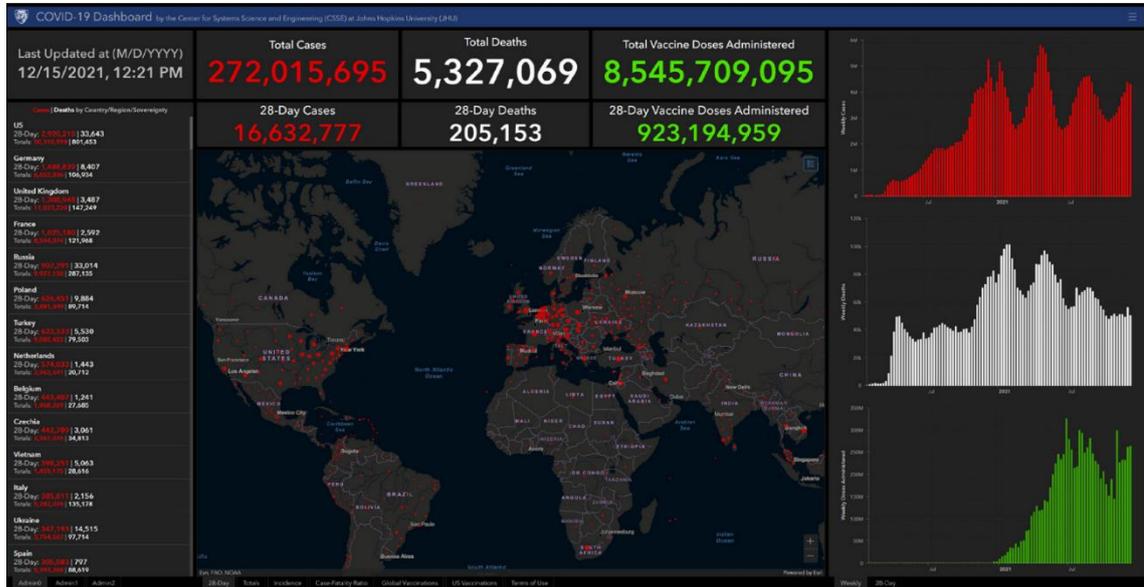


Figura 4 - Johns Hopkins COVID-19 Dashboard, que serviu para monitorizar a evolução da pandemia COVID-19 em tempo real, tornando-se numa referência global para o estudo da propagação e evolução do coronavírus.

Fonte: (Fu, 2018. p.5)

Do ponto de vista da tecnologia, estes sistemas envolvem uma série de componentes que trabalham em conjunto para atingir o objetivo final de fornecer um serviço capaz de partilhar e manipular informação geográfica. No *frontend* são utilizadas tecnologias como o HTML, CSS e JavaScript para criar a interface gráfica que permite aos utilizadores interagir com o sistema, e bibliotecas como o Leaflet ou OpenLayers para a criação de mapas. No *backend*, podem ser utilizadas uma grande variedade de tecnologias para a construção de uma API que permita a comunicação do *frontend* com o servidor. No caso do projeto gWater, e como já foi referido anteriormente, é utilizada a biblioteca Django REST Framework. Além disso, é necessário utilizar um servidor de mapas como o MapServer ou o GeoServer, sendo este último utilizado no projeto gWater. O *backend* é então responsável por fornecer os dados geográficos ao *frontend* através de *Web Services* utilizados para estes fins, como os *Web Map Service* (WMS) e *Web Feature Service* (WFS), que serão explicados de seguida.

A comunicação entre o *frontend* e *backend* é feita normalmente com recurso ao protocolo HTTP, com o *frontend* a fazer pedidos à API e esta devolvendo uma resposta que pode conter dados associados.

Os dados geográficos são normalmente guardados em bases de dados geoespaciais⁹, sendo necessário retirar esses dados da base de dados para serem utilizados por aplicações *web*. Com o intuito de facilitar a distribuição destes dados e torná-los mais acessíveis aos utilizadores, o Open Geospatial Consortium (OGC) definiu em 1999 um conjunto de padrões para a partilha destes dados. Foram definidos diversos serviços padronizados de forma a suportar a utilização de dados georreferenciados em qualquer plataforma conectada à internet (Peterson, 2012). Entre os serviços definidos destacam-se os *Web Map Service* e os *Web Feature Service*, uma vez que foram usados no trabalho desenvolvido:

- *Web Map Service* – Permite disponibilizar informação geográfica na forma de imagens georreferenciadas, sendo apenas permitida a leitura dos dados. Pode ser criado através de qualquer ficheiro *raster* ou vetorial. Quando os dados são requisitados é enviada uma imagem em formato JPEG ou PNG georreferenciada, onde cada pixel tem um valor RGB associado, ou seja, não são enviados os dados originais para o cliente.¹⁰
- *Web Feature Service* – Os *Web Feature Service* disponibilizam os dados geográficos em formato vetorial, permitindo o acesso direto aos mesmos. São criados através de dados vetoriais (pontos, linhas e polígonos). Quando um cliente efetua um pedido, são enviadas as geometrias e os atributos alfanuméricos num formato específico, como GeoJSON ou GML¹¹.

Os serviços WMS e WFS, são então tipos de *Web Service*, que permitem a comunicação e troca de informação geográfica.

Os WebSIG podem ser descritos como a interligação de cinco componentes diferentes, representados na Figura 5. Os dados (ex: formato *shapefile*, GeoJSON ou CSV) são utilizados para publicar *web layers*, que podem ser de vários tipos, incluindo *feature layers*, *tiled layers*, e as *layers* padrão do *Open Geospatial Consortium* (OGC), tais como WFS e WMS, que foram utilizadas para a construção da página de visualização e manipulação de dados do projeto gWater. Um exemplo de um *software* capaz de realizar este serviço é o GeoServer, também utilizado neste projeto. Uma vez

⁹ Bases de dados direcionadas para o armazenamento e manipulação de informação geográfica.

¹⁰ <https://www.ogc.org/standard/WMS/> (acedido em 4 de novembro, 2023)

¹¹ <https://www.ogc.org/standard/WFS/> (acedido em 4 de novembro, 2023)

publicadas as *layers*, estas podem ser utilizadas para construir *Web Maps*, que quando combinados com ferramentas de processamento de dados geográficos dão origem a uma aplicação WebSIG (Fu, 2018).

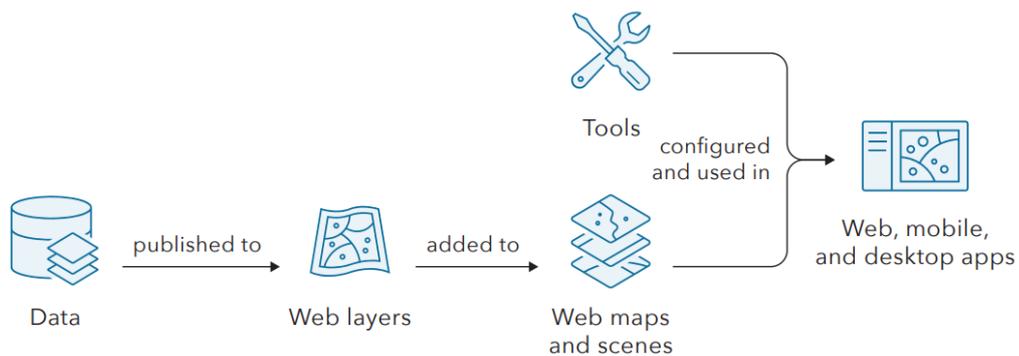


Figura 5 - Partes integrantes de um WebSIG.

Fonte:(Fu, 2018, p.11)

3. Monitorização da Qualidade da Água

3.1. Qualidade da Água

A poluição da água é um dos principais problemas ambientais da atualidade, representando nos dias de hoje uma ameaça à sobrevivência humana. A produção de alimentos em ambiente de aquicultura é diretamente influenciada pelas condições do estado da água, apresentando-se como um fator determinante na qualidade do produto final. Contudo, as ações de monitorização, minimização, controlo e mitigação dos problemas associados à sua qualidade, estão ainda longe de garantir o efeito desejado (Jin, 2022; Zhang et al., 2020).

A qualidade da água refere-se aos valores de diferentes variáveis num determinado momento. No contexto da aquicultura, os valores ótimos dessas variáveis podem ser diferentes conforme a finalidade e objetivo de cada produção, uma vez que cada espécie necessita de um ambiente favorável distinto, para que o seu crescimento e reprodução tenham a máxima eficiência possível, garantindo assim uma produção eficiente e sem perdas para os produtores. Estas variáveis podem ser de natureza física, como a temperatura; de natureza química, como o pH e o oxigénio dissolvido; e de natureza biológica, como a quantidade de clorofila existente nas plantas.

Tradicionalmente utilizam-se equipamentos flutuantes com sensores para fazer a medição da qualidade da água em tempo real. Apesar de estes equipamentos possuírem uma boa precisão, são bastante caros e a medição dessas variáveis num único ponto pode não representar corretamente toda a área de interesse desejada. Surge então, como complemento, a deteção remota, com custos potencialmente bastantes reduzidos para o utilizador, com capacidade de fornecer informação relativa a grandes áreas de interesse e com monitorização regular das mesmas (frequência) e a possibilidade de medição de diversos indicadores de qualidade da água, assumindo atualmente um lugar de destaque na monitorização da qualidade da água, nomeadamente em ambientes marinhos (Jin, 2022).

3.2. Harmful Algal Bloom

Os *Harmful Algal Bloom* (HAB) são eventos caracterizados pela proliferação de espécies de algas microscópicas tóxicas ou nocivas. Em determinados casos, e quando ocorrem em abundância, os seus pigmentos descoloram a água, isto é, alteram a sua cor natural. Contudo, existem espécies que não possuem altas concentrações celulares, não afetando a tonalidade da água, mas que devido

às toxinas produzidas também causam danos a humanos, peixes e outros animais (Anderson, 2009). Fatores como o aumento das atividades agrícolas são responsáveis pela introdução de nutrientes em corpos de água, afetando negativamente a biodiversidade e os *habitats* dos ecossistemas aquáticos, sendo o nitrogénio, que é amplamente utilizado para fins agrícolas, uma das principais fontes desses nutrientes. Esta introdução de nutrientes aumenta a produtividade dos sistemas aquáticos e aumenta conseqüentemente o crescimento e proliferação das respectivas algas prejudiciais à saúde dos seres humanos e dos ecossistemas (Karki et al., 2018). Estas ocorrências têm impactos económicos e ambientais bastante graves, particularmente em proliferações de algas dominadas por cianobactérias, que são normalmente a causa mais comum de florações e de toxinas prejudiciais. Apesar de haver HAB sem substâncias nocivas, estas podem levar na mesma à morte de muitos seres vivos nesses *habitats*, uma vez o número de seres vivos que realiza a respiração nessas águas aumenta e conseqüentemente os níveis de oxigénio na água diminuem de forma drástica (Carmichael & Boyer, 2016).

3.3. Monitorização com Dados Obtidos Através de Deteção Remota

A deteção remota é a ciência que permite obter informação crucial sobre a superfície da terra, sem estar em contacto com ela. Esta tecnologia permite obter dados ao longo do tempo que podem ter uma grande variedade de aplicações, tais como a monitorização do estado do tempo (meteorologia), estudo do uso e ocupação do solo, monitorização das mudanças provocadas pelas alterações climáticas, apoio às produções agrícolas, entre outros (Jensen, 2016). No contexto do projeto gWater, é importante destacar a monitorização do estado da água através de deteção remota, por exemplo, utilizando dados obtidos pelo programa Copernicus.

A aquisição de dados em deteção remota pode ser feita através de diversas técnicas, tais como a utilização sensores colocados em satélites, aviões ou veículos aéreos não tripulados (VANT). Apesar de diferentes, estas formas de recolha de dados em deteção remota funcionam da mesma forma, que consiste na captura de radiação eletromagnética através de um sensor, que pode possuir diversas bandas caso seja multiespectral ou hiperespectral, sendo capaz de medir a quantidade de radiação refletida ou emitida em cada banda, obtendo assim informações importantes acerca da assinatura espectral de um objeto, e permitindo distinguir objetos e obter informação acerca do seu estado. A Figura 6 descreve o processo de deteção remota desde a fonte de energia até à aplicação final dos dados obtidos, resultando em informação que irá servir de apoio a diversas áreas de trabalho (Ali, 2007).

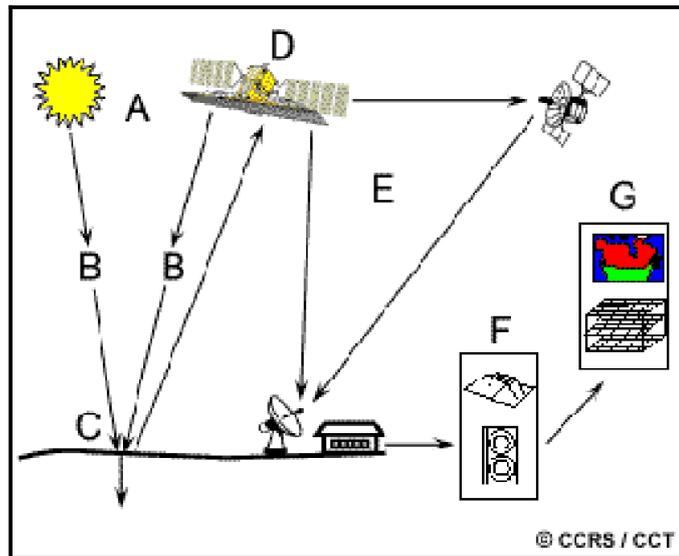


Figura 6 - Descrição do processo de detecção remota e os elementos que o compõem.

Fonte: National Resources Canada.

(A) fonte de energia/iluminação; (B) radiação; (C) interação da radiação com o solo; (D) sensor que mede a refletância de cada objeto nas diversas bandas; (E) transmissão; (F) recepção e processamento; (G) interpretação e análise.

O espectro eletromagnético é uma escala utilizada para caracterizar as diferentes interações da radiação eletromagnética com a matéria em função do seu comprimento de onda, que se pode dividir entre ondas de rádio, micro-ondas, infravermelhos, luz visível, ultravioleta, raios x e raios gama. Esta ordem representa a escala com início num maior comprimento de onda e menor frequência e orientada para um menor comprimento de onda e uma frequência maior (ver Figura 7).

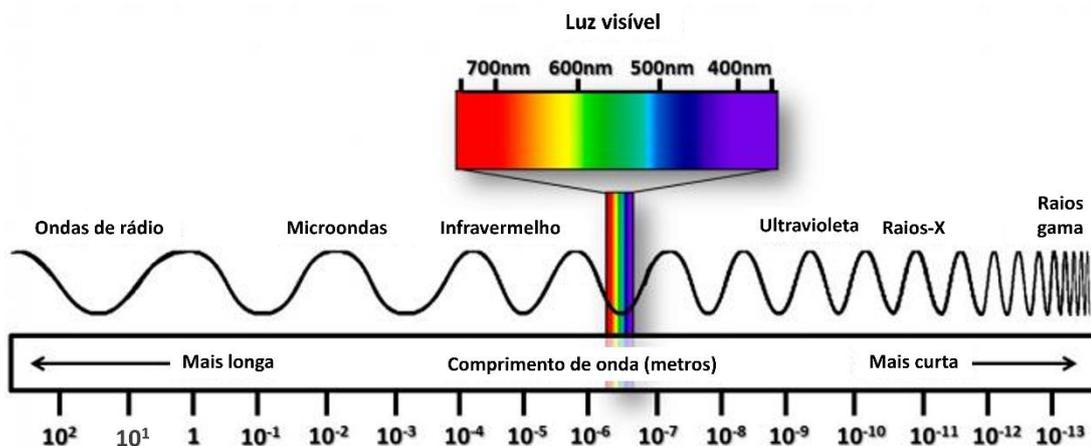


Figura 7 - Espectro eletromagnético.

Fonte: Adaptado de Climate Science Investigations (<https://www.ces.fau.edu/nasa/module-2/radiation-sun.php>)

Nenhuma tecnologia é perfeita, e como tal, podem existir erros no processo de aquisição dos dados que podem diminuir a sua qualidade e precisão. Os erros mais comuns durante este processo são erros radiométricos e geométricos, sendo por isso necessário fazer a sua correção. Os processos de correção destes erros são designados por operações de pré-processamento. Estes erros podem também ser classificados como internos ou externos, sendo internos quando têm origem no próprio sistema de deteção remota e externo quando é introduzido por elementos ou variáveis externas ao sistema de deteção remota, tais como a atmosfera, elevações do terreno ou declives (Jensen, 2016).

3.4. Dispositivos de Monitorização *in situ*

Podem ser obtidos dados para monitorização sistemática e automática da qualidade da água através da utilização de dispositivos de monitorização *in situ*, que são aparelhos que permitem fazer a medição dos parâmetros de qualidade da água no próprio local, de forma precisa e constante, através de diversos sensores capazes de fornecer dados em tempo real. Estes dispositivos podem ser facilmente adaptados para serem utilizados em diversos locais. Podem ser fixados em barcos, boias ou outro local subaquático, tal como uma jaula de aquicultura.

Como exemplo, vejamos o projeto Undersee da empresa MATEREO, cujo objetivo se assemelha bastante ao pretendido no projeto gWater, que é fazer a monitorização contínua da

qualidade da água, e que utiliza este tipo de dispositivos para fazer a recolha de dados necessários ao funcionamento do sistema. Os dispositivos utilizados neste projeto, chamados de “Undersee Devices”, fazem a medição de um vasto conjunto de parâmetros de qualidade da água. São eles a temperatura, o pH, o oxigénio dissolvido, a condutividade/salinidade, a turbidez e a clorofila-a.¹² Estes dados são transmitidos em tempo real para a *cloud* onde são tratados e armazenados.

Na Figura 8 são visíveis os componentes que compõem um Undersee device. Na Figura 9 podemos ver dois locais possíveis para o funcionamento deste aparelho, mostrando assim a sua versatilidade.

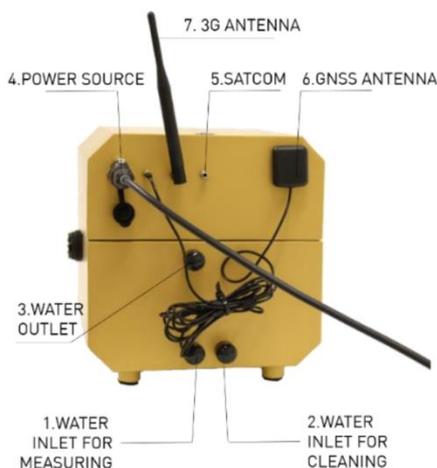


Figura 8 - Undersee device.

Fonte:

https://www.ipma.pt/pt/media/noticias/documentos/2019/UNDERSEE_IPMA_projectDescription.pdf

¹² Informação retirada de: <https://undersee.io/ferrybox/> (acedido a 6 de novembro, 2023); <https://business.esa.int/projects/undersee> (acedido a 6 de novembro, 2023); <https://business.esa.int/news/undersee-%E2%80%93-improved-decision-making-satellite-observations-and-forecasting-water-quality-changes-marine> (acedido a 6 de novembro, 2023).

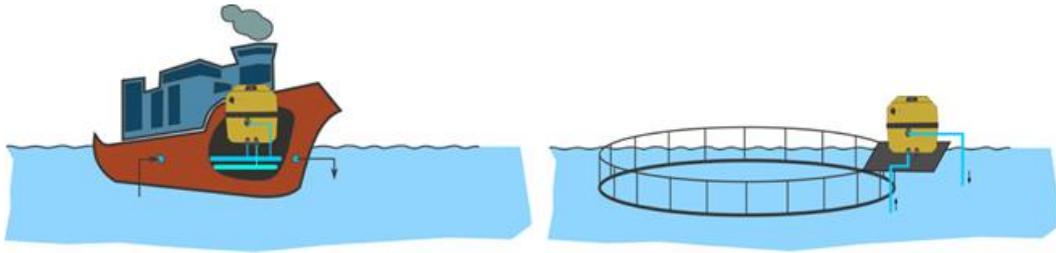


Figura 9 - Ilustração do funcionamento dos Undersee Devices em diferentes localizações.

Fonte: <https://business.esa.int/projects/undersee>

As ferryboxes¹³ tradicionais precisam de manutenção constante e por conseguinte intervenção humana, ao contrário do Undersee device, que oferece uma maior adaptabilidade, permitindo a sua utilização em qualquer barco, e ainda um sistema de autolimpeza que garante uma diminuição das necessidades de manutenção.

Os dados recolhidos pelos Undersee Devices podem ser visualizados e descarregados para utilização futura. Através destes dados, é possível construir um grande conjunto de gráficos e tabelas que podem contribuir para um aumento de eficiência nas tomadas de decisões¹⁴.

Apesar de esta funcionalidade não se encontrar presente no sistema gWater, que apenas utiliza dados provenientes de imagens de satélite, seria interessante fazer combinação com estes dois métodos de recolha de dados, uma vez que se podem complementar e tornar um sistema deste género mais robusto e com mais versatilidade, além do aumento de confiabilidade num sistema com estes dois métodos de recolha de dados, uma vez que seria possível confrontar os dados obtidos através do processamento de imagens de satélite, que muitas vezes pecam por falta de exatidão, com os dados recolhidos por este tipo de sensores *in situ*.

3.5. Monitorização da Qualidade da Água e o Papel dos WebSIG

A qualidade da água e os recursos hídricos estão a ser afetados devido ao aumento das pressões sobre os ecossistemas, causadas pelas alterações climáticas e pelas atividades humanas (Foley et al., 2005).

¹³ Uma FerryBox é um aparelho que efetua a medição de diversos parâmetros de qualidade da água, normalmente instalados em barcos ou navios.

¹⁴ <https://undersee.io/ferrybox/> (acedido a 6 de novembro, 2023)

No contexto de ambiente marinho, a monitorização em tempo real e previsão da qualidade da água é essencial para entender as dinâmicas e os processos físicos e químicos envolvidos (Aznar et al., 2016). Desta forma, essa monitorização serve de apoio à tomada de decisões de alto risco para a gestão costeira, desde a mitigação dos riscos naturais até ao apoio a diversas atividades económicas. Dentro das atividades económicas destaca-se a aquicultura, atividade em que é necessário garantir que os parâmetros utilizados para medir a qualidade da água estão dentro dos valores adequados para cada espécie, uma vez que cada espécie tem o seu ambiente marinho mais adequado para se reproduzir e crescer de forma mais saudável. Para fazer esta monitorização contínua da qualidade da água, é necessário recorrer a diversos métodos de recolha de dados, desde deteção remota até à utilização de sensores *in situ* (Pramana et al., 2021), que podem ser integrados num WebSIG. Para isso, é necessário que diferentes instituições, autoridades e comunidades consigam comunicar e partilhar dados de forma coordenada e eficiente. Os WebSIG apresentam-se como uma ferramenta de gestão e monitorização dos recursos hídricos através do acesso a informação de forma interativa e intuitiva (Herrera et al., 2021).

Quanto à utilização de imagens de satélite, é necessário ter em conta a periodicidade da aquisição de imagens, de maneira a ter imagens com bastante regularidade. É também importante ter atenção à resolução espacial e aos parâmetros considerados críticos e com necessidade de monitorização, uma vez que requerem diferentes bandas e diferentes combinações. Além disso, existem também serviços como o CMEMS que fornece dados tratados de forma gratuita utilizando imagens de vários satélites do programa Copernicus.

Como exemplo da utilidade destas imagens, pode-se destacar as imagens de concentração de clorofila, que quando em grandes quantidades indica uma maior concentração de nutrientes e por sua vez uma maior produção de algas nocivas.

3.6. Exemplos de WebSIG que Realizam a Monitorização da Qualidade da Água

Nesta secção são identificadas três aplicações web que fazem a monitorização contínua da qualidade da água, de maneira a fornecer informação útil aos seus utilizadores. Uma vez que estas aplicações possuem uma página em que é possível visualizar informação geográfica, são adequadas para utilizar como referência de aplicações existentes com as funcionalidades que se pretendem implementar no sistema gWater. Serve também esta análise para identificar as semelhanças e diferenças entre estas aplicações, permitindo identificar falhas ou funcionalidades em falta e deste modo servir de guia para a implementação destas funcionalidades no sistema gWater.

3.6.1. MyOcean

Na plataforma MyOcean do programa Copernicus, é possível monitorizar parâmetros como a temperatura, salinidade e clorofila. Os dados utilizados nesta plataforma são provenientes de satélites, sistemas de medição terrestres e de modelos numéricos¹⁵.

Nesta plataforma é possível adicionar diversas camadas com diversos parâmetros e navegar ao longo de uma linha do tempo, de maneira a obter informação sobre a evolução ou visualizar dados de um período específico (ver Figura 10). É também possível visualizar a informação de determinada variável num determinado ponto do mapa, sendo apenas necessário um clique num determinado local. Além disso, é possível definir intervalos de corte de maneira a filtrar os dados e visualizar dados para o intervalo de valores especificado.

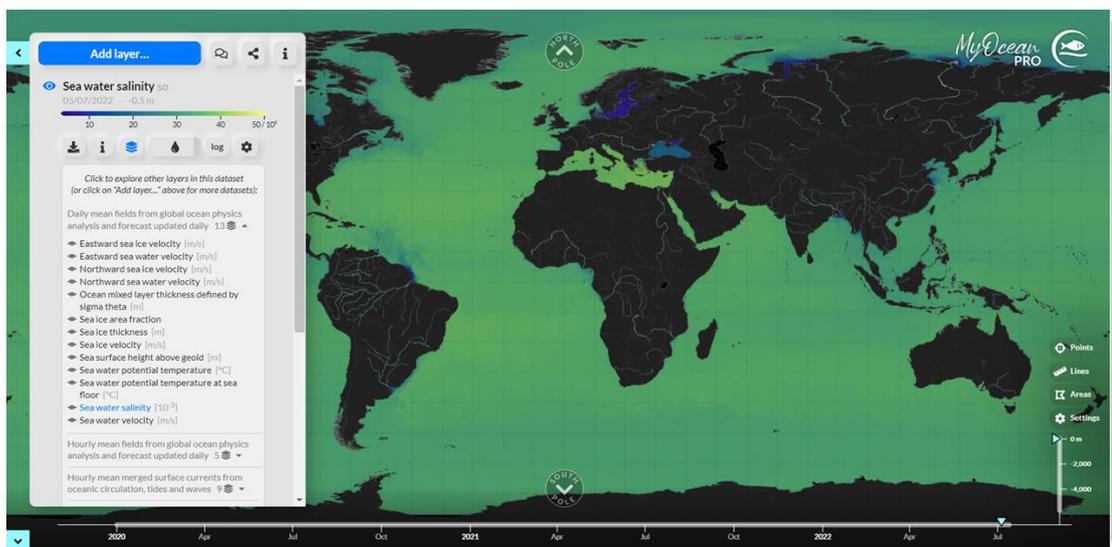


Figura 10 - Página principal da plataforma MyOcean.

Fonte: <https://data.marine.copernicus.eu/viewer>

¹⁵ <https://marine.copernicus.eu/pt/acesso-dados/visualizador-meu-oceano> (acedido em 9 de junho, 2023)

3.6.2. WorldView

A plataforma WorldView da NASA também permite adicionar ao mapa diversas camadas com os mais variados parâmetros e visualizar as mesmas. Permite igualmente visualizar essas variáveis em diferentes períodos, através de uma linha do tempo (ver Figura 11). Nesta plataforma não é possível visualizar informação acerca da variável em qualquer ponto do mapa através de um clique. Contudo, ao passar o ponteiro do rato por cima dos dados é apresentado um intervalo de valores por baixo da barra com a escala de cores da camada, que se encontra do lado esquerdo da página. Por último, esta plataforma destaca-se por ter imagens com maior resolução.

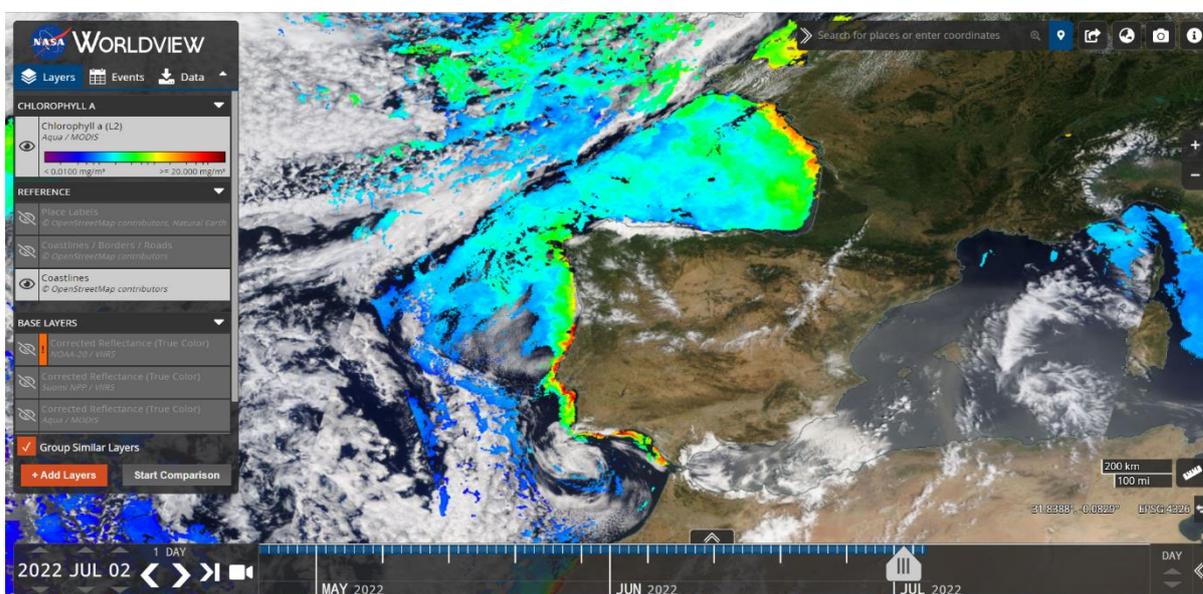


Figura 11 - Página principal da plataforma WorldView.

Fonte: <https://worldview.earthdata.nasa.gov/>

3.6.3. COASTNET

Na plataforma COASTNET é possível visualizar dados acerca de diversos parâmetros obtidos através de imagens de satélite e clicar em qualquer lugar do mapa para abrir um *popup* com mais informação. Além disso, também é possível visualizar dados recolhidos por sensores *in situ* em diversos locais na foz dos rios Mondego, Tejo e Mira, complementando assim os dados obtidos através de satélites e tornando esta plataforma bastante completa (ver Figura 12). Desta forma, é possível confrontar os dados das duas fontes existentes e obter informação mais fidedigna e precisa.

Este é um bom exemplo de como complementar o sistema gWater através da utilização deste tipo de sensores.

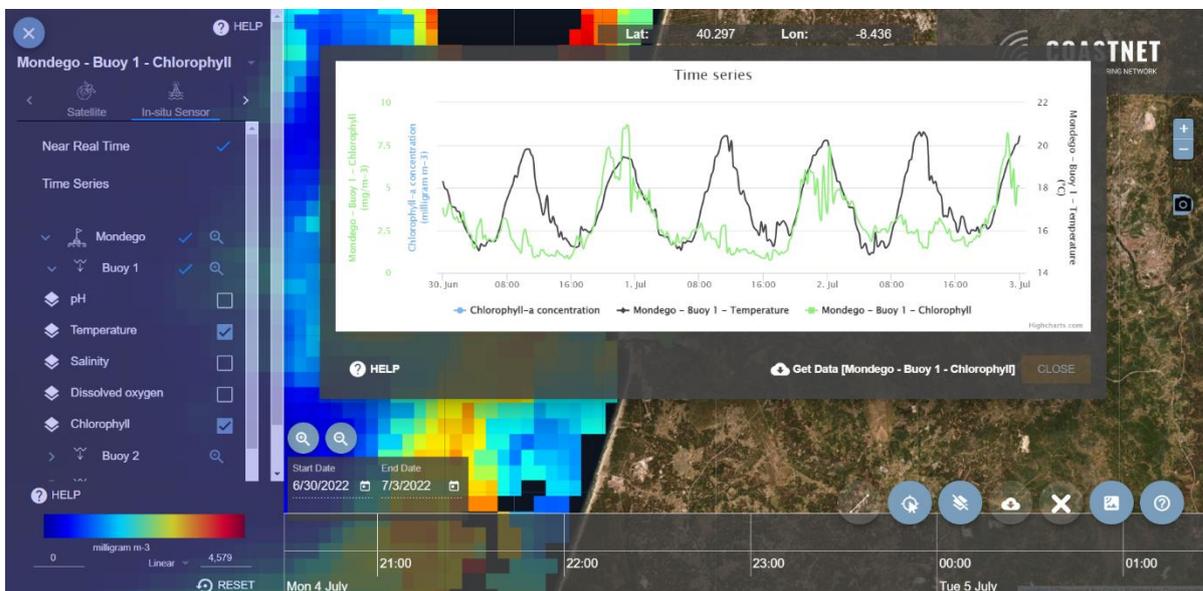


Figura 12 - Página principal da plataforma COASTNET.

Fonte: <http://geoportal.coastnet.pt/>

3.7. Análise e Comparação dos Sistemas

Através da análise das aplicações mencionadas na secção anterior, é possível identificar algumas semelhanças, tais como a possibilidade de adicionar camadas ao mapa relativas a parâmetros como a clorofila ou a temperatura, e a existência da funcionalidade de percorrer uma linha do tempo. No entanto, existem também algumas diferenças que alteram a interação do utilizador com a plataforma, como é o caso dos *pop-ups* com informação extra acerca dos valores dos parâmetros no pixel que foi clicado. Outro aspeto importante está relacionado com o tamanho do pixel, que torna a leitura dos dados para um determinado local mais ou menos precisa. Quanto à variação temporal dos dados recolhidos, que já foi referida acima, todas as plataformas incorporam uma barra de navegação temporal, que permite a observação da variação dos diversos parâmetros ao longo do tempo de forma bastante intuitiva. No caso do sistema COASTNET, existe uma funcionalidade que permite visualizar valores de parâmetros recolhidos por sensores presentes na água, tais como a temperatura, o pH, a salinidade e oxigénio dissolvido.

No MyOcean, também é possível visualizar informação com origem em sensores *in-situ*, mas o seu funcionamento é mais complexo. Além disso, o CMEMS tem outra plataforma dedicada apenas a observações *in situ* chamada **In Situ TAC**¹⁶. Esta funcionalidade não existe na plataforma WorldView, que apenas permite adicionar camadas ao mapa relativas aos diversos parâmetros, não apenas parâmetros relacionados com a monitorização da água, mas também diversos parâmetros relacionados com outras categorias como a qualidade do ar, secas, entre outros, o que torna esta plataforma bastante completa de uma forma geral. Além disso, na plataforma WorldView existe a possibilidade de alteração dos *thresholds* de forma a filtrar os dados visualizados, semelhante à funcionalidade do visualizador do MyOcean com o mesmo objetivo.

Quanto à possibilidade de execução de ferramentas de geoprocessamento de forma a manipular os dados, nenhuma destas plataformas contempla essa funcionalidade. No entanto, existem funcionalidades como a medição de distâncias e áreas no MyOcean e no WorldView.

Desta forma, o sistema gWater poderá colmatar as limitações destas plataformas quanto à manipulação dos dados, através da execução de ferramentas dos mais variados tipos, podendo gerar novos dados através dos dados originais, e acrescentando assim informação que poderá ser útil na tomada de decisões. A título de exemplo, podem existir ferramentas que criam polígonos representativos de áreas com valores elevados de uma determinada variável. Ou caso o sistema seja adaptado para outra finalidade, poderá existir uma ferramenta que faça a deteção de determinada espécie exótica invasora. Assim o sistema gWater poderá ir mais além em relação aos exemplos apresentados acima, tornando-se numa ferramenta poderosa no contexto da visualização e manipulação de informação geográfica.

¹⁶ <https://marineinsitu.eu/dashboard/>

4. Drivers Arquiteturais

A especificação dos drivers arquiteturais tem como objetivo definir e detalhar os requisitos associados ao sistema gWater. Além disso, são apresentados os constrangimentos técnicos, e ainda os atributos de qualidade. Os requisitos funcionais pretendem descrever o comportamento funcional do sistema através de uma lista de funcionalidades esperadas, ou seja, que o sistema deverá conseguir realizar, ajudando a traduzir as necessidades do sistema em funcionalidades que deverão estar disponíveis para o utilizador final. Estes requisitos estão associados a funcionalidades gerais, descrevendo apenas operações que os utilizadores devem conseguir realizar na interface gráfica do sistema. Um requisito funcional poderá corresponder a diversas operações do sistema. Um exemplo disso é a gestão das camadas dos dados do CMEMS (requisito funcional), que implica um conjunto de diferentes operações. Este nível de detalhe será exibido nos diagramas e tabelas de casos de uso, que se encontram no Apêndice A, onde será detalhada a forma como o sistema reage à interação com os utilizadores. Nos diagramas e casos de uso, serão também detalhadas todas as operações que a API consegue executar, tais como a criação de uma *store* no GeoServer, que pode ser feita sem recurso a interface gráfica, que é o que acontece aquando da recolha automática dos dados. No entanto, é assumido que todas as operações são efetuadas através da interface gráfica do sistema. Os casos de uso presentes em cada diagrama estão identificados, permitindo relacionar com a respetiva tabela de caso de uso, que terá toda a informação necessária acerca do funcionamento dessa operação. As tabelas de caso de uso mostram quais os atores responsáveis por realizar a operação, os requisitos funcionais e *endpoints* associados, qual o objetivo da operação, as pré-condições, o cenário principal e o cenário alternativo. As tabelas com os *endpoints* encontram-se no Apêndice B.

Quanto aos constrangimentos técnicos, estes dizem respeito a limitações como linguagem de programação utilizada, *hardware* utilizado, bibliotecas e *frameworks* disponíveis, entre outros. Os atributos de qualidade são características que o sistema deve apresentar, tais como eficácia ou eficiência. Podem também estar associados a aspetos como a segurança ou escalabilidade do sistema.

4.1. Requisitos Funcionais

Nesta secção, é apresentada a Tabela 1, que contém todos os requisitos funcionais elaborados no contexto do sistema gWater. A tabela contém quatro colunas. A primeira identifica o requisito através de um ID, que será utilizado nas tabelas de caso de uso para fazer a associação com os requisitos. A segunda, apresenta a descrição do requisito, sendo aqui onde se indica a funcionalidade que deverá existir no sistema. A terceira indica o nível de necessidade do requisito, uma vez que existem requisitos fundamentais para o funcionamento do sistema e outros que são um complemento ao sistema mais básico, por isso é feita a distinção entre requisitos necessários e desejados. A quarta coluna é utilizada para fazer um comentário que complementa a descrição do requisito, fornecendo informação extra para o entendimento do mesmo.

Tabela 1 - Requisitos funcionais do sistema gWater.

ID Requisito Funcional	Descrição do Requisito Funcional	Necessário/ Desejado	Comentário
RF001	O sistema deve ter uma página de login que permita aos utilizadores fazer a sua autenticação.	Necessário	Esta página não terá a opção de registo. Este será feito apenas por um superutilizador (dono do sistema) através da página de gestão do sistema.
RF002	O utilizador é redirecionado para a página principal do sistema depois de fazer a autenticação.	Necessário	
RF003	O utilizador deverá conseguir fazer <i>logout</i> do sistema.	Necessário	
RF004	O utilizador deverá conseguir alterar a sua palavra-passe.	Necessário	
RF005	O utilizador deverá conseguir recuperar a sua palavra-passe.	Necessário	Deverá existir um botão na página de login com o texto “Forgot your password?”
RF006	O utilizador deve conseguir navegar por entre um catálogo de dados existentes no sistema.	Necessário	Cada utilizador deverá ter apenas acesso a dados relacionados com as áreas de interesse associadas a si.
RF007	O sistema deverá disponibilizar automaticamente um conjunto de camadas associadas a dados CMEMS, Sentinel-2 e áreas de algas.	Necessário	As áreas com ocorrências de algas são obtidas através dos dados do Sentinel-2.
RF008	O catálogo de dados deverá estar estruturado em árvore com os seguintes níveis: 1) Área de interesse; 2) Fonte dos dados; 3) Variável (apenas no caso dos dados do CMEMS)	Necessário	
RF009	O sistema deverá adicionar dados ao catálogo de forma sistemática, através da recolha e tratamento de dados de forma automatizada.	Necessário	Dados CMEMS, Sentinel-2 e Algas
RF010	O utilizador deverá conseguir adicionar qualquer conjunto de dados presentes no catálogo ao Map Layers.	Necessário	O Map Layers é onde estão as camadas que podem ser adicionadas ao mapa, utilizadas para a construção de gráficos e utilizadas como input na execução de ferramentas.
RF011	O utilizador deverá conseguir remover qualquer camada presente no Map Layers.	Necessário	
RF012	O utilizador deverá conseguir ativar as camadas presentes no Map Layers através de uma caixa de seleção presente no início do nome de cada camada.	Necessário	A camada fica visível no mapa.
RF013	O utilizador deverá conseguir desativar qualquer cada que esteja marcada como ativada no Map Layers.	Necessário	A camada é removida do mapa.
RF014	O utilizador deve conseguir utilizar camadas relacionadas com dados CMEMS ou Sentinel-2 para a criação de histogramas.	Necessário	
RF015	O utilizador deve conseguir definir o número de intervalos do histograma. No caso de serem dados do Sentinel-2, deve poder inserir um valor máximo para o histograma.	Necessário	O sistema deve indicar o valor mínimo e máximo que o utilizador pode fornecer para o número de intervalos.
RF016	O histograma deve ser interativo.	Desejado	Ao passar com o cursor por cima do gráfico, devem surgir os valores exatos de cada barra.
RF017	O utilizador deve conseguir utilizar camadas relacionadas com dados CMEMS para a criação de gráficos de dispersão.	Necessário	
RF018	O sistema deverá verificar que as camadas introduzidas nos respetivos campos para a criação do gráfico de dispersão são da mesma área de interesse.	Necessário	Não faz sentido de outra forma.
RF019	O utilizador deverá conseguir fazer upload de um <i>shapefile</i> com os pontos necessários para a construção do gráfico de dispersão.	Necessário	O ficheiro tem de estar no formato <i>zip</i> .

ID Requisito Funcional	Descrição do Requisito Funcional	Necessário/ Desejado	Comentário
RF020	O utilizador deverá conseguir criar os pontos no mapa como alternativa ao upload de um ficheiro com pontos criados previamente.	Desejado	
RF021	O utilizador deve conseguir utilizar camadas relacionadas com dados CMEMS para a criação de gráficos de evolução temporal.	Necessário	
RF022	O utilizador deve conseguir ver a legenda de cada camada presente no Map Layers.	Necessário	A legenda fica por baixo do item da camada no Map Layers. O utilizador deve clicar no botão de expansão para expandir a legenda. Caso clique novamente, a legenda é ocultada. O comportamento padrão ao adicionar uma camada ao Map Layers é legenda oculta.
RF023	O utilizador deverá conseguir alterar a simbologia de uma camada CMEMS presente no Map Layers.	Necessário	Deve escolher a escala de cores desejada, assim como o número de intervalos e o método para o cálculo dos valores desses intervalos.
RF024	O utilizador deverá conseguir alterar a ordem das camadas presentes no Map Layers.	Necessário	O utilizador seleciona e arrasta uma camada para a posição desejada. Essa alteração é refletida no mapa, uma vez que camadas no topo ficarão sobrepostas a camadas que estão mais em baixo no Map Layers.
RF025	O utilizador deverá conseguir alterar o basemap,	Desejado	Deve haver diversas alternativas ao basemap padrão.
RF026	O utilizador deverá conseguir alterar a composição de bandas de uma camada Sentinel-2 presente no Map Layers.	Necessário	Devem existir pelo menos duas composições possíveis. Composição a cor natural e infravermelhos.
RF027	O utilizador deverá conseguir navegar por uma lista das ferramentas disponíveis no sistema.	Necessário	Essa lista deve estar estruturada em árvore, onde o primeiro nó representa um conjunto de ferramentas, que ao ser expandido revela as ferramentas presentes nesse grupo.
RF028	O utilizador deverá conseguir escolher uma ferramenta existente na lista de ferramentas.	Necessário	Basta clicar no item da lista.
RF029	O sistema deverá abrir uma janela relacionada com a ferramenta que um utilizador escolha.	Necessário	Essa janela contém todos os campos necessários para a execução da ferramenta. O utilizador deve preencher todos os campos.
RF030	O utilizador deverá ser capaz de executar a ferramenta após preencher todos os campos necessários.	Necessário	O utilizador clica no botão "Run".
RF031	O sistema deverá informar o utilizador acerca do progresso da execução da ferramenta.	Necessário	O sistema emite notificações ao longo do processo.
RF032	O sistema deverá adicionar os dados obtidos através da execução de uma ferramenta ao catálogo de dados.	Necessário	Deve haver uma fonte de dados chamada "Tool Results"
RF033	O utilizador deverá conseguir eliminar os resultados obtidos através da execução de uma ferramenta.	Necessário	
RF034	O utilizador deverá conseguir aceder a uma página de gestão da sua conta.	Desejado	
RF035	O utilizador deverá conseguir aceder a uma página de gestão dos seus dados.	Necessário	
RF036	O utilizador deverá conseguir adicionar uma camada sua ao Map Layers.	Desejado	Esta camada não estará presente no catálogo de dados fornecido pelo sistema.
RF037	O utilizador deverá conseguir adicionar um script personalizado á lista de ferramentas.	Desejado	Essa script fica disponível num novo nó de nível 1 com nome à escolha do utilizador.
RF038	O superutilizador deverá conseguir fazer a gestão de todos os utilizadores através da página de gestão associada à sua conta.	Necessário	A página de gestão do superutilizador contém mais funcionalidades relativamente à página do utilizador comum.

ID Requisito Funcional	Descrição do Requisito Funcional	Necessário/ Desejado	Comentário
			A gestão dos utilizadores inclui: 1) Adicionar utilizador; 2) Editar dados do utilizador; 3) Eliminar utilizador.
RF039	O superutilizador deverá conseguir fazer a gestão dos estilos existentes no sistema.	Necessário	Estes estilos estão relacionados com as camadas CMEMS. O superutilizador deverá conseguir ver, adicionar, editar e eliminar estes estilos.
RF040	O utilizador deverá ser capaz de fazer <i>zoom to layer</i> numa determinada camada	Desejado	
RF041	O superutilizador deverá conseguir fazer a gestão das variáveis utilizadas no sistema.	Necessário	Relacionado com dados CMEMS. O superutilizador deverá adicionar, editar e remover variáveis.
RF042	O superutilizador deverá conseguir fazer a gestão das áreas de interesse existentes no sistema através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover áreas de interesse.
RF043	O superutilizador deverá conseguir fazer a gestão dos <i>workspaces</i> do GeoServer através da sua página de gestão ou diretamente na interface gráfico do GeoServer.	Necessário	Deverá conseguir obter, adicionar e remover um <i>workspace</i> .
RF044	O superutilizador deverá conseguir fazer a gestão das <i>stores</i> do GeoServer através da sua página de gestão ou diretamente na interface gráfico do GeoServer.	Necessário	Deverá conseguir obter, adicionar e remover uma <i>store</i> .
RF045	O superutilizador deverá conseguir fazer a gestão das camadas do GeoServer através da sua página de gestão ou diretamente na interface gráfico do GeoServer.	Necessário	Deverá conseguir obter, adicionar, editar e remover uma camada.
RF046	O superutilizador deverá conseguir fazer a gestão dos estilos do GeoServer através da sua página de gestão ou diretamente na interface gráfico do GeoServer.	Necessário	Deverá conseguir obter, adicionar, editar e remover um estilo.
RF047	O superutilizador deverá conseguir fazer a gestão de todos os mosaicos do Sentinel-2 através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover um mosaico Sentinel-2.
RF048	O superutilizador deverá conseguir fazer a gestão de todas as camadas do CMEMS através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover uma camada CMEMS. Isto está associado à tabela que guarda todos os dados associados com uma camada. Isto não afeta os dados existentes no GeoServer, porém os utilizadores deixam de conseguir pedir os dados ao GeoServer caso seja eliminado um registo nesta tabela, uma vez que é removida do catálogo.
RF049	O superutilizador deverá conseguir fazer a gestão de todos os <i>Algae Bloom</i> Datasets através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover um <i>Algae Bloom</i> Dataset.
RF050	O superutilizador deverá conseguir fazer a gestão de todas as camadas <i>Algae Bloom</i> através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover uma camada <i>Algae Bloom</i> .
RF051	O superutilizador deverá conseguir fazer a gestão de todas as camadas Sentinel-2 através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover uma camada Sentinel-2.
RF052	O superutilizador deverá conseguir fazer a gestão de todas as legendas CMEMS através da sua página de gestão.	Necessário	Deverá conseguir obter, adicionar, editar e remover uma legenda CMEMS.

4.2. Constrangimentos Técnicos

Os constrangimentos técnicos estão associados a restrições impostas durante o processo de desenvolvimento, podendo ter diversas naturezas, tais como os recursos existentes, tempo dedicado ao desenvolvimento, experiência ou até decisões tomadas anteriormente. De seguida, são identificados os constrangimentos técnicos existentes no contexto do sistema gWater. Estes são apresentados nas Tabelas 2 a 6, e estão agrupados em função da sua origem. Cada tabela apresenta uma descrição do constrangimento, seguida da justificação da existência do mesmo. Por último, é apresentado o nível de impacto do constrangimento e uma possível estratégia para a sua mitigação.

Tabela 2 - Constrangimento técnico relacionado com a utilização do Angular.

C1 Tecnologias	
Descrição	Tecnologias utilizadas para a criação da aplicação de cliente, API e servidor de mapas.
Justificação	As tecnologias utilizadas correspondem às tecnologias que o líder de equipa domina, não dispondo de tempo para aprofundar outras alternativas. Além disso, o Django é baseado em Python, o que permite a utilização de diversas bibliotecas Python utilizadas para lidar com dados geoespaciais.
Impacto	Baixo.
Estratégia de Mitigação	Aprofundamento dos conhecimentos técnicos associado a essas tecnologias.

Tabela 3 - Constrangimento técnico relacionado com a aquisição dos dados do Sentinel-2.

C2 Aquisição de Dados Sentinel-2	
Descrição	Não é possível descarregar dados ao mesmo tempo.
Justificação	O Sentinel Hub impõe limites de <i>downloads</i> de dados através da API em contas gratuitas.
Impacto	Médio.
Estratégia de Mitigação	Não existe.

Tabela 4 - Constrangimento técnico relacionado com a equipa de desenvolvimento.

C3 Equipa de desenvolvimento	
Descrição	Número limitado de elementos na equipa.
Justificação	A equipa é constituída por apenas dois elementos, sendo que um deles não pode dedicar 100% do seu tempo ao desenvolvimento do sistema.
Impacto	Alto. A carga de trabalho será demasiada para duas pessoas.
Estratégia de Mitigação	Priorizar tarefas e funcionalidades essenciais para o funcionamento do sistema.

Tabela 5 - Constrangimento técnico relacionado com o hardware.

C4 Hardware	
Descrição	Limitações do <i>hardware</i> utilizado.
Justificação	O desenvolvimento deve ser feito com recurso a equipamentos pessoais, uma vez que não foi fornecido equipamento ao mestrando para a realização do estágio.
Impacto	Alto. O equipamento pessoal apresenta limitações a nível de desempenho, podendo reduzir a eficiência ao longo do desenvolvimento, que pode apresentar tempos de processamento bastantes longos. Além disso o hardware poderá não conseguir aguentar com todo o <i>software</i> necessário para o desenvolvimento do sistema a correr ao mesmo tempo.
Estratégia de Mitigação	Otimização do <i>software</i> . O código deve ser otimizado de maneira a aumentar a sua eficiência e reduzir assim a utilização dos recursos disponíveis.

Tabela 6 - Constrangimento técnico relacionado com o servidor.

C5 Servidor	
Descrição	Não existe servidor para testar o sistema em produção.
Justificação	Não será disponibilizado qualquer servidor durante a realização deste estágio, e não será adotada nenhuma alternativa que implique algum tipo de custo monetário.
Impacto	Alto.
Estratégia de Mitigação	Utilização de contentores Docker e máquinas virtuais para simular o ambiente de produção.

4.3. Atributos de Qualidade

Os atributos de qualidade, também designados por requisitos não funcionais, são critérios que definem como é que o sistema se deve comportar a nível qualitativo. São características que o sistema deve ter e não funcionalidades concretas. Estes atributos são fundamentais para garantir a eficácia e eficiência do sistema.

De seguida são apresentados os principais atributos de qualidade do sistema gWater através de tabelas individuais. Cada tabela corresponde a um atributo de qualidade geral, sendo detalhado em uma ou mais propriedades. Essas propriedades contêm uma descrição do requisito não funcional e um possível cenário onde é apresentada a relevância do mesmo.

Tabela 7 - Atributo de qualidade relacionado com a segurança do Sistema gWater.

Segurança			
Confidencialidade	Descrição	O sistema gWater deve ser um sistema privado. Dessa forma, o sistema deve ser acedido apenas por utilizadores registados no mesmo, através de um sistema de autenticação.	
	Cenário	Fonte	Utilizador não autorizado.
		Estímulo	O utilizador tenta aceder a uma página do sistema sem estar autenticado.
		Resposta	O sistema redireciona o utilizador para a página de <i>login</i> e não apresenta dados ou informações restritas.
	Resultado Esperado	O sistema nega o acesso a qualquer utilizador não registado.	
Integridade	Descrição	O sistema gWater deve apresentar precisão e consistência. Os seus dados não devem ser alterados de forma acidental ou inadequada, assim como através de ações maliciosas.	
	Cenário	Fonte	Superutilizador.
		Estímulo	O utilizador faz um pedido ao servidor para obter algum tipo de informação.
		Resposta	A informação recebida pelo utilizador corresponde exatamente à informação que existe no sistema.
	Resultado Esperado	O sistema apresenta sempre a informação completa enviada pelo servidor.	

Tabela 8 - Atributo de qualidade relacionado com a interoperabilidade do Sistema gWater.

Interoperabilidade			
Integração com outros sistemas	Descrição	Um utilizador do sistema pretende fazer o <i>upload</i> de um <i>shapefile</i> com pontos para a criação de um gráfico de dispersão através das imagens existentes no sistema.	
	Cenário	Fonte	Utilizador do sistema.
		Estímulo	Um utilizador do sistema pretende fazer o <i>upload</i> de um <i>shapefile</i> criado com recurso a um <i>software</i> SIG com pontos para a criação de um gráfico de dispersão usando as imagens existentes no sistema.
		Resposta	O sistema gWater é capaz de ler este tipo de ficheiro e realizar operações com o mesmo.
	Resultado Esperado	A integração entre o sistema gWater e um qualquer <i>software</i> SIG ocorre sem problemas, garantindo a possibilidade de partilha de dados entre os dois sistemas.	
Padrões	Descrição	O sistema deve adotar formatos ou padrões de dados abertos de maneira a facilitar a partilha dos mesmos.	
	Cenário	Fonte	Equipa de desenvolvimento.
		Estímulo	A equipa decide procurar alternativas ao GeoServer mantendo a compatibilidade com os padrões OGC.
		Resposta	A equipa consegue testar um servidor de mapas alternativo mantendo a adoção dos mesmos padrões.
	Resultado Esperado	É possível testar soluções alternativas mantendo a conformidade com os padrões da OGC, reduzindo assim a complexidade e tempo que uma operação desta natureza envolve.	

Tabela 9 - Atributo de qualidade relacionado com a precisão do Sistema gWater.

Precisão			
Alta Precisão nas Medições	Descrição	O sistema gWater deve assegurar que as análises da qualidade da água realizadas com recurso aos dados do CMEMS e Sentinel-2 estão de acordo com os dados disponibilizados.	
	Cenário	Fonte	Utilizador do sistema.
		Estímulo	O utilizador necessita de garantir a robustez do sistema, no sentido de que não há erros nos dados disponibilizados.
		Resposta	O sistema gWater recolhe dados de duas fontes diferentes, nomeadamente do CMEMS e imagens do Sentinel-2, permitindo a comparação das duas fontes de dados.
	Resultado Esperado	As medições do sistema gWater são bastante confiáveis, no sentido que se baseiam nos dados disponíveis.	

Tabela 10 - Atributo de qualidade relacionado com a usabilidade do sistema gWater.

Usabilidade			
Aprendizagem	Descrição	O sistema deve assegurar a facilidade de utilização para um utilizador que o está a utilizar pela primeira vez.	
	Cenário	Fonte	Utilizador do sistema.
		Estímulo	Adicionar uma camada ao Map Layers.
		Resposta	O utilizador consegue adicionar uma camada presente no catálogo de dados ao Map Layers.
	Resultado Esperado	O sistema tem uma interface gráfica que permite aos utilizadores realizar as diferentes tarefas sem ter experiência prévia na utilização do mesmo.	
Eficiência na Utilização	Descrição	Rapidez com que os utilizadores conseguem executar diversas tarefas após compreenderem como funciona.	
	Cenário	Fonte	Utilizador do sistema.
		Estímulo	Um utilizador do sistema pretende realizar uma tarefa que requer um conjunto de passos até atingir o objetivo final.
		Resposta	O utilizador consegue realizar a operação desejada através de poucos cliques.
	Resultado Esperado	O sistema gWater oferece uma interface gráfica amigável e bem estruturada, permitindo a realização de diferentes operações no menor tempo possível e com o menor número de cliques.	
Estética	Descrição	O design do sistema deve ser agradável e contribuir para uma boa experiência de utilização.	
	Cenário	Fonte	Utilizador do sistema.
		Estímulo	O utilizador pretende visualizar um histograma associado a uma determinada camada.
		Resposta	O utilizador visualiza e interpreta o histograma interativo de forma fácil, conseguindo visualizar todos os valores relevantes no contexto da visualização deste tipo de gráfico.
	Resultado Esperado	O sistema deve apresentar informação de forma simples e didática, permitindo a fácil interpretação por parte dos utilizadores.	

Tabela 11 - Atributo de qualidade relacionado com a flexibilidade e adaptabilidade do Sistema gWater.

Flexibilidade e Adaptabilidade			
Adaptação a Novas Tecnologias	Descrição	Implementação de novas funcionalidades e melhorias no sistema de forma eficiente.	
	Cenário	Fonte	Equipa de desenvolvimento.
		Estímulo	Utilização de novas tecnologias no contexto da deteção remota e monitorização da qualidade da água.
		Resposta	O sistema gWater é construído com uma arquitetura flexível que garante a facilidade de integração de novas tecnologias.
	Resultado Esperado	O sistema é capaz de se manter atualizado no contexto da monitorização da qualidade da água.	
Personalização	Descrição	Facilidade de personalização do sistema para atender a necessidades específicas dos utilizadores.	
	Cenário	Fonte	Utilizador do sistema.
		Estímulo	Um utilizador do sistema tem necessidades específicas quanto à monitorização da qualidade da água.
		Resposta	O sistema gWater permite aos utilizadores definir alertas caso sejam ultrapassados determinados valores considerados críticos relacionados com variáveis à sua escolha existentes no sistema.
	Resultado Esperado	O sistema gWater permite a sua personalização de modo a atender às necessidades de todos os utilizadores, contribuindo para a sua usabilidade e satisfação dos utilizadores. Os utilizadores obtêm rapidamente informações relevantes para si.	

5. Arquitetura do Sistema

Os sistemas de *software* são construídos para satisfazer as necessidades de organizações ou de um negócio. Para atingir esse fim, é necessário definir uma arquitetura que consiga transportar a ideia e os objetivos definidos até ao resultado. A arquitetura de *software* pode ser desenhada, documentada e implementada de modo a reduzir a complexidade da tradução dos objetivos em sistemas que satisfaçam esses objetivos. Consiste na estrutura que define os componentes de um sistema de *software*, as suas definições e as relações entre si. Apresenta também a relação com outros sistemas, externos ao sistema em desenvolvimento, detalhando os protocolos de comunicação utilizados. Estas relações mostram de que forma o trabalho individual de cada elemento do sistema contribui para a concretização do objetivo final (Bass et al., 2012). Neste capítulo, será apresentada a arquitetura de *software* do sistema gWater seguindo a lógica do modelo C4.

5.1. Modelo C4

O modelo C4, tal como indica o nome, é composto por quatro camadas de diagramas: o diagrama de contexto, diagrama de contentores, diagrama de componentes e código. A palavra-chave que permite entender esta visualização da arquitetura é *zoom*. É exatamente este o conceito aplicado no modelo C4, onde cada diagrama apresenta um nível de detalhe diferente, isto é, correspondem a diferentes níveis de abstração. Ao contrário de outras abordagens para a criação deste tipo de estruturas, como Unified Modeling Language (UML), não existe uma notação ou simbologia padrão para o uso deste modelo, sendo apenas necessário apresentar coerência ao longo de todos os diagramas elaborados, onde cada elemento que seja repetido em diagramas diferentes tenha a mesma forma e aspeto, e ainda fornecer uma legenda de modo a aumentar a sua clareza (Bonocore & Singhchawla, 2022).

No contexto do sistema gWater, são apresentados de seguida três diagramas tendo como base o modelo C4. Será apresentado em primeiro lugar o **diagrama de contexto**, de maneira a fornecer uma vista de alto nível do sistema, uma vez que apresenta apenas a relação do sistema como um todo com os atores intervenientes e com sistemas externos. O diagrama seguinte, **diagrama de contentores**, apresenta um maior nível de detalhe, fazendo um *zoom* para dentro do sistema gWater, e expondo assim os diversos contentores que o compõem e as relações entre eles. Além disso, mostra as tecnologias utilizadas em cada um dos contentores, assim como as

tecnologias utilizadas nas relações. O terceiro diagrama, **diagrama de componentes**, mostra o que existe dentro de cada contentor, aumentando assim o nível de detalhe em relação ao diagrama anterior. Tal como no diagrama de contentores, no diagrama de componentes também são descritas as tecnologias utilizadas em cada componente, e apresentadas as relações existentes entre eles. O último nível, **diagrama de código**, apresenta o nível máximo de detalhe possível neste modelo de arquitetura de *software*, sendo normalmente omitido, uma vez que neste nível de detalhe existem alterações significativas, traduzindo-se muitas vezes em dificuldades em manter este diagrama atualizado em relação às alterações do código¹⁷. O diagrama de código será omitido neste relatório, uma vez que não se justifica um nível de detalhe demasiado elevado. Estes tipos de diagramas são muitas vezes obtidos automaticamente através de IDEs (Integratd Development Environment) utilizados para desenvolver o código (Bonocore & Singhchawla, 2022).

A Figura 13 mostra a legenda dos diagramas apresentados nas secções seguintes.



Figura 13 - Legenda dos diagramas de contexto, contentores e componentes.

5.1.1. Diagrama de Contexto

O diagrama de contexto (Figura 14), tal como já foi referido, tem como objetivo apresentar uma vista de alto nível do sistema gWater, uma visão geral que permite mostrar quais os utilizadores e sistemas externos que interagem com o sistema. Neste diagrama, o sistema gWater surge no centro, como um todo, existindo três relações com utilizadores e três relações com sistemas externos. Os utilizadores que utilizam o sistema são do tipo “Superuser” e “Client User”, sendo

¹⁷ <https://c4model.com/> (acedido em 8 de novembro, 2023)

que apenas existe um do primeiro tipo e os restantes são utilizadores do segundo tipo. O utilizador do tipo “**Superuser**” é quem gere o sistema, aquele que possui todas as permissões, podendo fazer todo o tipo de operações e manipulações no sistema. Este utilizador é o responsável pelo processo de desenvolvimento e implementação do sistema, e por isso apenas poderá existir um utilizador deste tipo, uma vez executa operações críticas, como apagar todo o conteúdo de uma tabela. Quanto ao utilizador do tipo “**Client User**”, este é quem utiliza o sistema. Este utilizador possui permissões para visualizar e manipular informação geográfica tendo em conta as áreas de interesse associadas a si, tendo a possibilidade de criar gráficos que permitam fazer diversas análises de evolução e tendência dos dados, assim como a correlação entre diferentes conjuntos de dados. Este tipo de utilizador faz ainda a gestão dos parâmetros de qualidade da água e define os respetivos intervalos de corte necessários para a criação de alertas por parte do sistema.

Ainda dentro do diagrama de contexto, são apresentados os sistemas externos com os quais o sistema gWater interage, contribuindo para toda a lógica do sistema e fornecendo dados essenciais para o seu funcionamento, acrescentando assim valor ao sistema como um todo. São eles, a API do CMEMS e a API do Sentinel Hub, que permitem obter imagens pré-processadas relativas aos diversos parâmetros de qualidade da água e imagens do Sentinel-2, respetivamente.

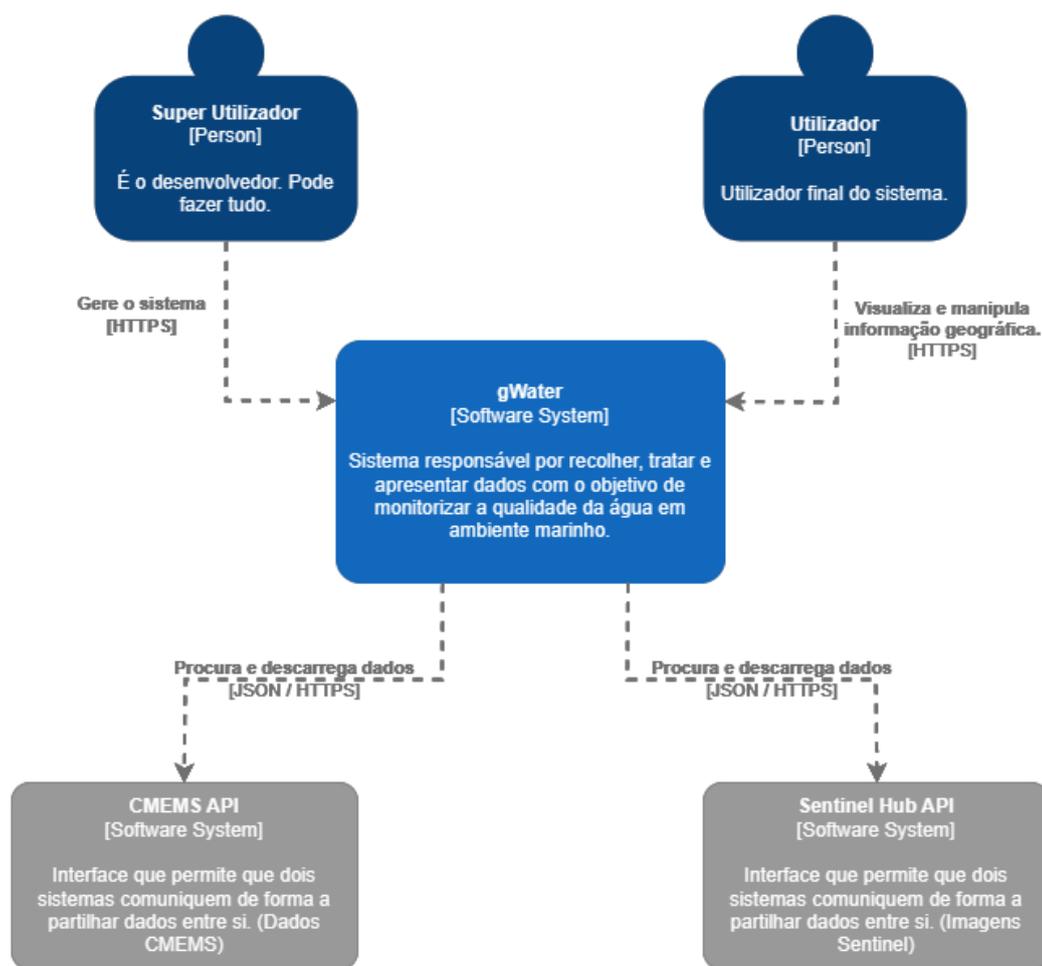


Figura 14 - Diagrama de Contexto do Sistema gWater

5.1.2. Diagrama de Contentores

No segundo nível é aumentado o nível de detalhe, através do diagrama de contentores (Figura 15), que mostra o que existe dentro do sistema gWater, dividido por aplicações ou módulos. Este diagrama mostra como esses módulos interagem entre si e de que forma os utilizadores interagem com o sistema através desses módulos.

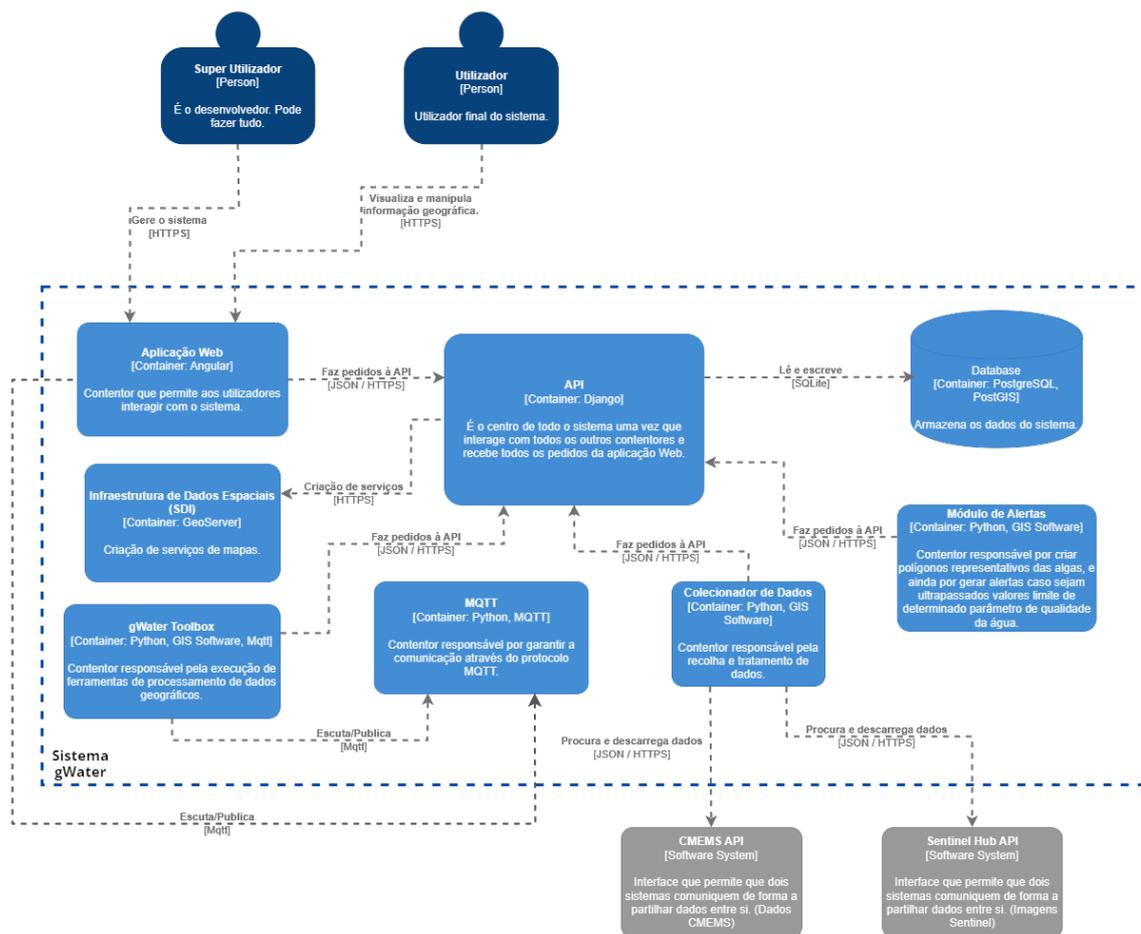


Figura 15 - Diagrama de Conteúdos do Sistema gWater.

A Figura 15 mostra que os utilizadores interagem com o sistema através da aplicação *web* desenvolvida com Angular. Este módulo, **Aplicação Web**, é responsável por toda a interface gráfica do sistema, apresentando todo o conteúdo existente no sistema aos utilizadores. Este módulo comunica com a API do sistema através de pedidos HTTP, que têm como objetivo a obtenção, alteração, envio ou eliminação de dados. A API devolve uma resposta conforme o solicitado. A aplicação *web* comunica ainda com o módulo MQTT através do protocolo MQTT, uma vez que foi esta a solução encontrada para obter o progresso da execução de uma ferramenta, processo este que pode ser demorado, permitindo assim aos utilizadores continuarem a utilizar o sistema enquanto acompanham o progresso da execução da mesma.

O módulo **MQTT** é, neste caso, apenas responsável pela comunicação da aplicação *web* com o módulo gWater Toolbox, no contexto da execução de ferramentas. Contudo, no caso da utilização de sensores *in situ* que façam a medição dos parâmetros que medem a qualidade da água, possibilidade de complemento do sistema já referida neste relatório, e que implicaria o acrescento

de mais um sistema externo a estes diagramas, a sua ligação ao sistema gWater seria precisamente com este contentor, uma vez que este tipo de comunicação é feita normalmente com recurso ao protocolo MQTT, devido às suas características, que serão detalhadas no capítulo que descreve a implementação do sistema. Nesse caso, seria necessário existir uma comunicação entre o contentor MQTT e o contentor Colecionador de Dados, uma vez que este último seria o responsável por subscrever o tópico necessário para receber as publicações dos dispositivos *in situ* e por fazer o devido tratamento desses dados antes de fazer a comunicação com a API.

O módulo **Colecionador de dados** é o responsável pela recolha dos dados fornecidos pelos sistemas externos. Este módulo é constituído por uma série de *scripts* que descarregam, tratam e carregam os dados na base de dados, através de chamadas à API. O tratamento realizado neste contentor tem como objetivo fazer um pré-processamento dos dados de maneira a atenderem aos requisitos do sistema gWater, além de serem executados uma série de processos que permitem a apresentação destes dados no sistema gWater de forma gráfica, num mapa de visualização de dados. Para tal, é necessária a criação de serviços de mapas no GeoServer e dos respetivos estilos. Estes processos serão explicados no capítulo sete, que detalha os processos de implementação do sistema.

O contentor **gWater Toolbox** é responsável pela execução de ferramentas de geoprocessamento, sendo constituído por um conjunto de *scripts* que garantem a execução desses processos e fazem ainda a comunicação com a API e também diretamente com a aplicação *web* através do protocolo MQTT, de forma a garantir informação aos utilizadores acerca do progresso destes processos.

No contentor **Módulo de Alertas** são gerados os alertas representativos de eventos de algas, através de um conjunto de *scripts* que verificam os intervalos de corte definidos pelos utilizadores para cada variável de qualidade da água e emitem um alerta caso esses valores sejam ultrapassados. Além disso, são criados polígonos representativos das áreas afetadas por essas alterações de valores, que poderão ser adicionados ao mapa sob a forma de *Web Feature Service*.

O contentor **Infraestrutura de Dados Espaciais** tem como objetivo o fornecimento de dados para os mapas existentes no sistema gWater, através do GeoServer, o servidor de mapas utilizado neste sistema. A comunicação da aplicação *web* com este módulo é feita através de chamadas à API por parte da aplicação *web*, sendo depois feita uma comunicação da API com este módulo, garantindo assim que os utilizadores tenham acesso visual às imagens existentes no sistema através de um mapa.

Por último, o contentor **API** atua como o cérebro deste sistema, recebendo chamadas de diversos contentores e processando-as de acordo com a lógica implementada nos diversos *endpoints*.

Este contentor possui uma ligação com a base dados, uma vez que é responsável por enviar e requisitar os dados da mesma, garantindo assim um fluxo adequado de transmissão de dados dentro do sistema.

Nos contentores acima apresentados, apenas o contentor Módulo de Alertas ficou de fora do âmbito deste estágio, sendo implementado por outro membro da equipa do projeto gWater.

5.1.3. Diagrama de Componentes

Por último, são apresentados dois diagramas de componentes, revelando as componentes e a suas relações dentro dos dois contentores mais complexos e que merecem um maior nível de detalhe que revele o seu funcionamento e a sua complexidade. Os dois diagramas apresentados de seguida revelam em detalhe o que acontece dentro do contentor “**Aplicação Web**” e “**API**”. No entanto, serão apenas destacadas as componentes principais que demonstrem a lógica de funcionamento dentro destes dois contentores, permitindo ao leitor entender o diagrama sem grandes dificuldades. De notar que os componentes do contentor “Aplicação Web” aqui definidos não estão associados diretamente a componentes Angular existentes na aplicação web, mas sim a um agrupamento de funções gerais que permita mostrar de forma simplificada como funciona a aplicação.

Na Figura 16, é apresentada uma visualização do que acontece dentro do contentor “Aplicação Web”, permitindo visualizar os componentes que o integram e saber de que forma interage com o exterior.

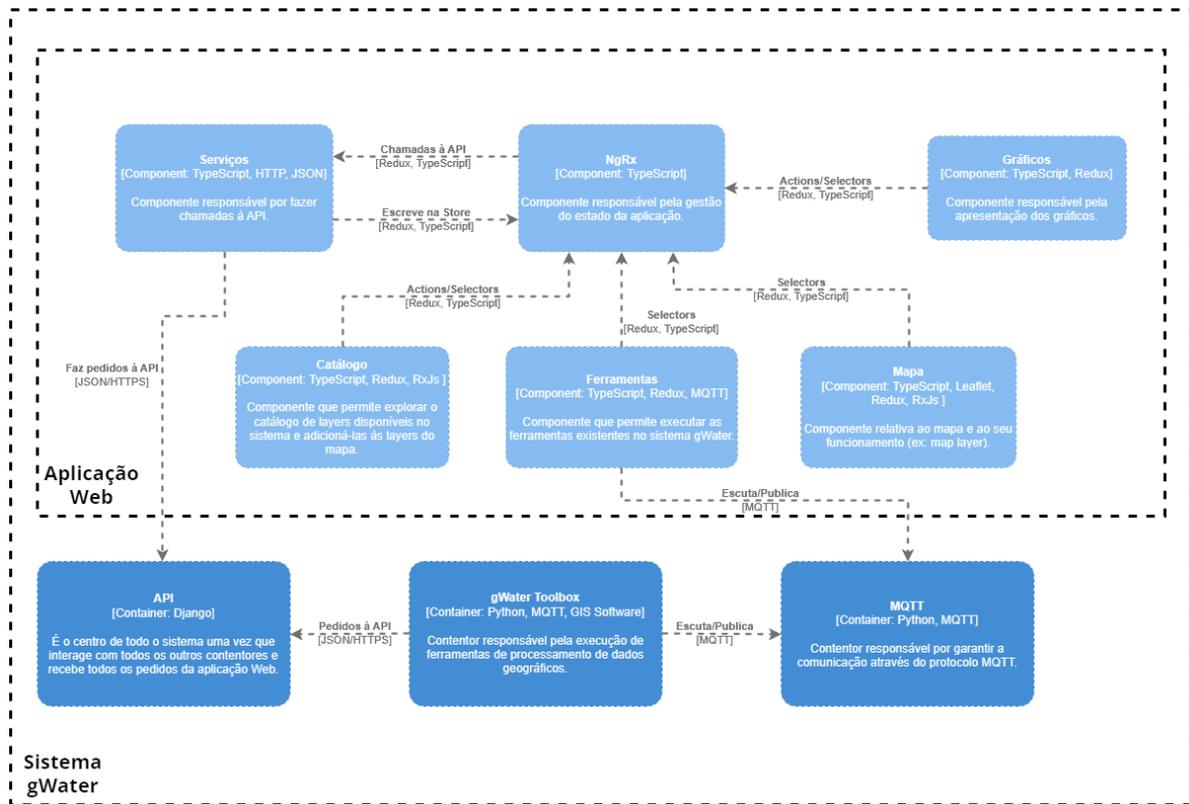


Figura 16 - Diagrama de componentes do contêiner "Aplicação web" do sistema gWater.

Neste diagrama foram deixados de fora os utilizadores, uma vez que o foco do mesmo é a lógica de funcionamento dentro do sistema gWater. O contêiner “Aplicação Web” foi dividido em 6 componentes. São elas a componente “NgRx”, “Serviços”, “Catálogo”, “Ferramentas”, “Mapa” e “Gráficos”.

A componente “**NgRx**” atua como peça central no fluxo de funcionamento da aplicação *web*, uma vez que permite o armazenamento centralizado do estado dentro da aplicação, permitindo a sua utilização em toda a aplicação, sendo necessário utilizar ações para provocar mudanças nesse estado, refletindo-se em toda a aplicação, independentemente de serem componentes Angular separadas, evitando assim a utilização de serviços para a partilha de dados entre componentes. Esta componente está por isso ligada a todas as outras componentes do sistema através de *actions* e *selectors*, que permitem a alteração ou obtenção de uma determinada propriedade do estado, tal como já foi explicado no segundo capítulo. A sua relação com a componente “Serviços” é diferente, uma vez que esta relação apenas se verifica no contexto de chamadas à API, ou seja, quando é necessário obter dados de fontes externas à aplicação *web*, normalmente à API, e cuja resposta poderá provocar alterações no estado, dependendo do sucesso ou insucesso da solicitação. Neste

contexto, são utilizados os *effects* NgRx, que tal como já foi referido no segundo capítulo, permitem lidar com os serviços da aplicação.

A componente “**Serviços**”, tal como já foi referido, é a responsável pela comunicação da aplicação *web* com a API, através de chamadas utilizando o protocolo HTTP. Neste contexto, foi adicionado o contentor externo “API” neste diagrama, de maneira a demonstrar esta lógica de funcionamento e a relação entre estes dois contentores, com maior detalhe do lado do contentor “Aplicação Web”.

A componente “**Catálogo**”, é a responsável por permitir ao utilizador explorar os dados geográficos existentes no sistema, podendo adicioná-las às *layers* do mapa, que podem também ser utilizadas para a criação de gráficos que permitem fazer diversas análises, comportamento que será explicado mais á frente. Esta componente encontra-se ligada à componente “NgRx”, uma vez que seleciona o estado, o que permite obter a lista das *layers* disponíveis e que são carregadas no estado logo após o *login* do utilizador, de acordo com as áreas de interesse associadas à sua conta, e também através do despacho de ações, que ocorrem quando é necessário adicionar uma *layer* ao chamado “Map Layers”.

A componente “**Ferramentas**” foi desenhada para permitir a escolha entre um conjunto de ferramentas, tal como acontece nos *softwares* SIG tradicionais. Após a escolha de uma ferramenta são apresentados os diversos parâmetros necessários para dar início à sua execução, aos quais o utilizador tem de atribuir um valor, sendo normalmente a escolha de uma ou mais *layers* do “Map Layers” o primeiro parâmetro necessário. Esta componente encontra-se ligada à componente “NgRx” uma vez que é necessário selecionar o estado para obter a lista de *layers* presentes no “Map Layers”. Além disso, encontra-se ligada ao contentor “MQTT”, sendo esta a outra ligação existente do contentor “Aplicação Web” com um contentor exterior. Esta ligação existe devido à necessidade de haver uma comunicação constante entre o contentor “gWater Toolbox” e o contentor “Aplicação Web”, que tal como já foi referido anteriormente e será ainda detalhado mais à frente neste relatório, é feita com recurso ao protocolo MQTT. A utilização deste protocolo permite a ligação contínua necessária neste contexto, possibilitando a apresentação de notificações do progresso de execução de uma ferramenta, fornecendo assim informação útil aos utilizadores e permitindo desta forma continuar a utilizar a aplicação enquanto aguardam pelo final da execução, do qual serão notificados. O contentor “MQTT” serve por isso de intermediário nesta relação entre os contentores “gWater Toolbox” e a componente “Ferramentas” do contentor “Aplicação Web”. A presença do contentor “gWater Toolbox” neste diagrama é justificada apenas pelo complemento da explicação feita acima, uma vez que não interage diretamente com o contentor descrito neste diagrama, que é a “Aplicação Web”.

A componente “**Mapa**” é uma peça essencial neste sistema, uma vez que é ela que permite mostrar os dados aos utilizadores, garantindo assim a funcionalidade de visualização de dados geográficos num mapa, e permitindo a interação direta com o mesmo. Além disso, permite ao utilizador definir como quer visualizar os dados, através da alteração da simbologia e definição do número de intervalos que pretende ter em cada camada em formato *raster*. Permite também a visualização de dados vetoriais, associados aos polígonos gerados aquando da execução da ferramenta que permite identificar algas através das imagens Sentinel-2. De notar que esta componente não se refere propriamente ao mapa em si, mas também ao “Map layers”, uma vez que está diretamente associado ao mapa e às suas funcionalidades. Esta componente encontra-se ligada à componente “NgRx”, uma vez que seleciona o estado e faz o despacho de ações com o objetivo de alterar propriedades desse estado.

Por último, a componente “Gráficos” é a responsável pela criação e visualização dos gráficos associados aos dados geográficos. O sistema gWater permite a criação de forma dinâmica de histogramas, gráficos de dispersão e gráficos de evolução temporal. Através destes gráficos, o utilizador obtém informações essenciais acerca da distribuição, evolução e relação entre os dados, permitindo fazer análises que ajudem na tomada de decisões, através da deteção de tendências e relações entre as oscilações dos diferentes parâmetros. Esta componente, tal como todas as anteriores, encontra-se ligada à componente “NgRx”, uma vez que seleciona o estado da aplicação e faz o despacho de ações que podem provocar a sua alteração.

De seguida, é apresentado na Figura 17 o diagrama de componentes do contentor “API”, que mostra de forma aprofundada como este funciona e como interage com outros contentores.

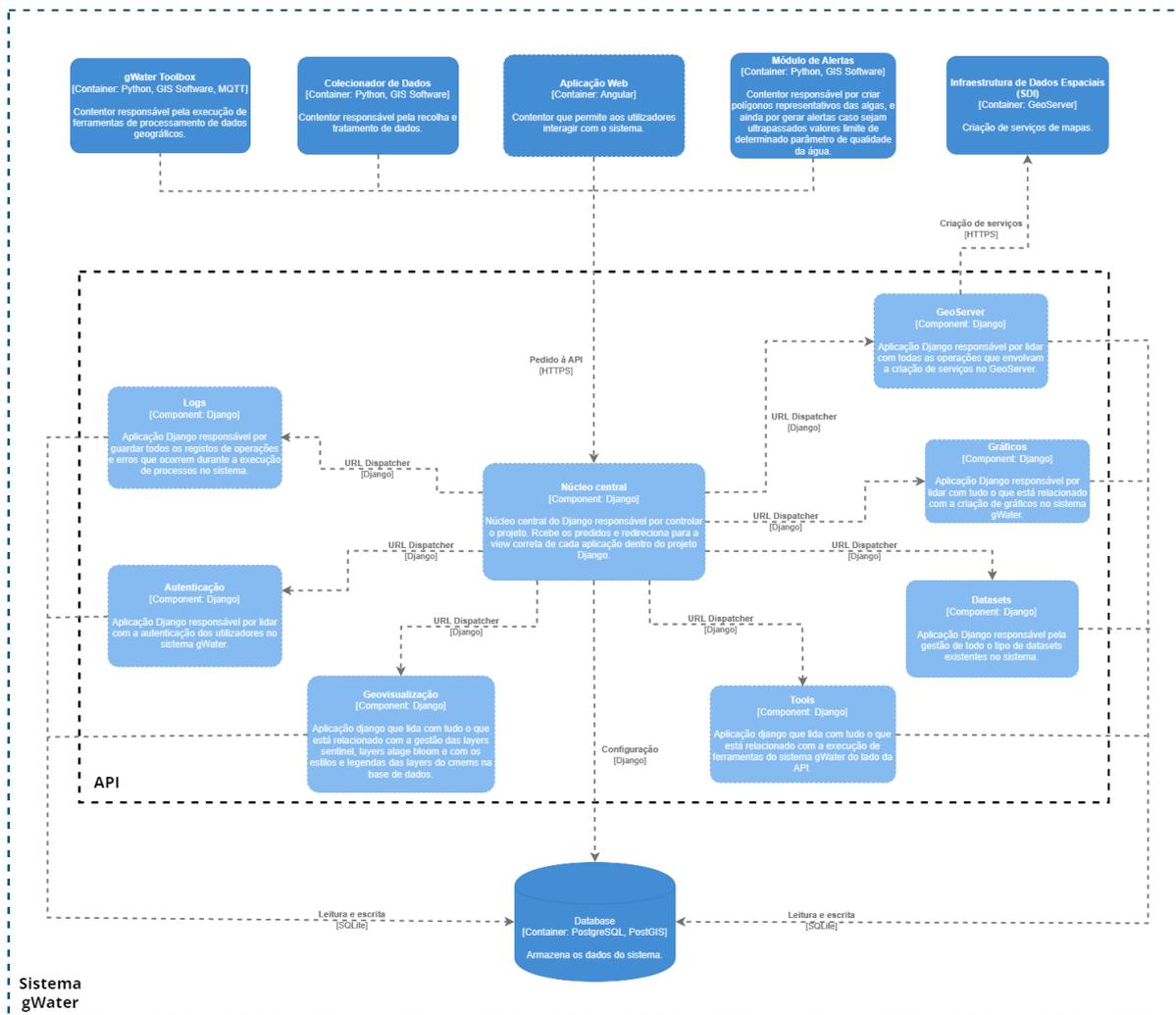


Figura 17 - Diagrama de componentes do contentor "API" do sistema gWater.

Tal como acontece com o diagrama de componentes do contentor “Aplicação Web”, também no diagrama de componentes do contentor “API” foram deixados de fora os utilizadores, uma vez que estes não interagem diretamente com a API, que é o foco principal deste diagrama.

No centro do diagrama surge o chamado “Módulo Central”, que não é mais que o projeto principal, onde se encontram os ficheiros “settings.py”, “manage.py”, “wsgi.py”, “urls.py”, entre outros. Estes ficheiros são fundamentais para o funcionamento de um projeto Django, sendo que iremos dar especial atenção ao ficheiro “urls.py”. Este ficheiro é responsável por fazer o roteamento dentro do projeto, ou seja, ela recebe todos os pedidos feitos à API por parte dos restantes contentores e direciona esses pedidos para a aplicação correspondente com base no seu URL, que por sua vez está associado a uma *view* específica no qual se encontra a lógica do *endpoint*, que vai ser responsável por devolver uma resposta a esse pedido. Neste diagrama, todas as componentes, à exceção da central, representam aplicações Django individuais, uma vez que esta

divisão padrão na lógica de funcionamento do Django é adequada para apresentar uma vista de arquitetura deste nível, mostrando a interação de cada aplicação com os restantes elementos dentro e fora do contentor “API”.

Todos os componentes comunicam com a base de dados através de operações de leitura e escrita, e todos comunicam com a componente central uma vez que dela recebem os pedidos feitos à API, sendo esta operação de redireccionamento conhecida como “URL dispatcher”. O módulo central, apesar de não realizar operações de leitura e escrita na base de dados, é o responsável pela sua configuração. A componente “GeoServer”, responsável por lidar com todas as operações que envolvam a criação de serviços no GeoServer e dos respetivos estilos, tem uma ligação extra, que é a ligação ao contentor “Infraestrutura de Dados Espaciais”, uma vez que é necessário existir esta comunicação para permitir a criação desses serviços no GeoServer, e assim permitir a visualização de dados por parte dos utilizadores.

De notar que todos os pedidos à API são protegidos, uma vez que todos os *endpoints* requerem autenticação do utilizador, verificando ainda qual o papel do mesmo, uma vez que o utilizador comum terá algumas limitações no contexto de realização de operações no sistema comparativamente ao superutilizador, que tal como já foi referido anteriormente possui permissões para realizar todas as operações disponíveis em todos os *endpoints*.

6. Tecnologias Utilizadas

A escolha das tecnologias utilizadas no projeto gWater foi um pouco condicionada desde o início, tal como foi referido na secção do quarto capítulo que aborda os constrangimentos técnicos. Contudo, foi feita uma revisão das mesmas de maneira a compreender o porquê destas escolhas prévias e perceber se existem alternativas mais adequadas para este projeto.

As tecnologias utilizadas podem ser divididas em diversas categorias, como tecnologias de *frontend*, tecnologias de *backend*, tecnologias de *web-mapping*, base de dados, virtualização, entre outras. Porém, nesta secção são apenas abordadas as tecnologias de *frontend*, tecnologias de *backend* e a tecnologia utilizada para fazer a comunicação entre os *scripts* de execução das ferramentas e a aplicação de cliente, de forma a evitar chamadas constantes à API.

6.1. Frontend

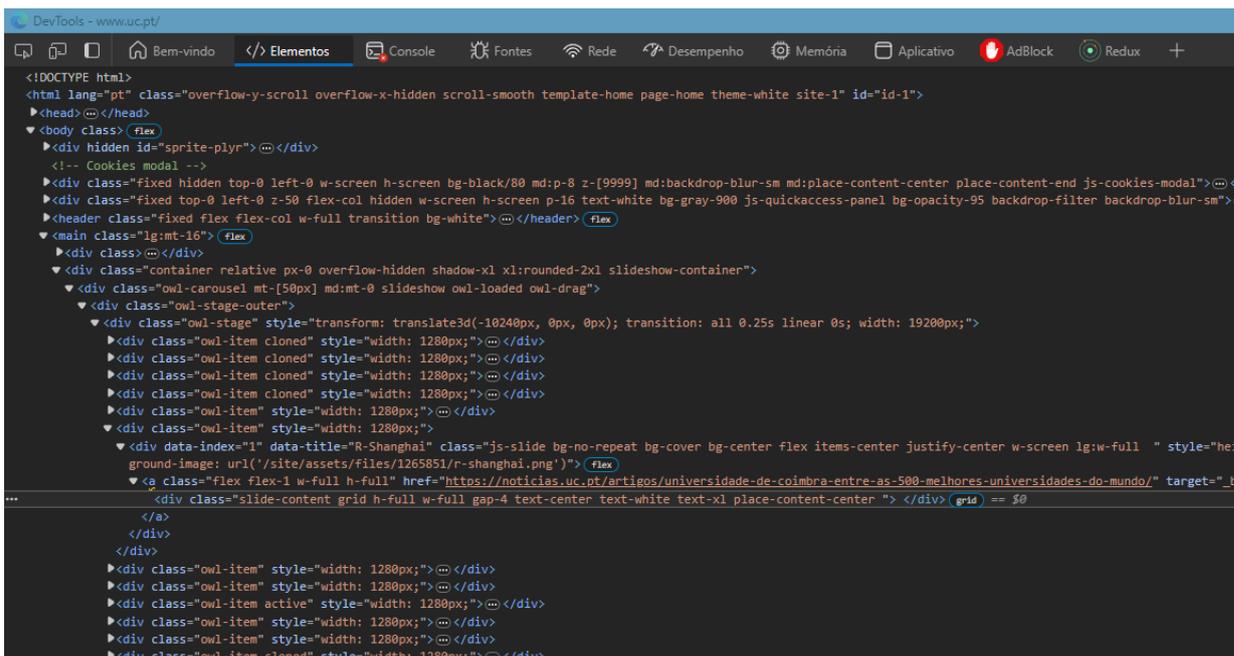
O desenvolvimento *frontend*, também conhecido como desenvolvimento do lado cliente, permite aos utilizadores interagirem com qualquer website ou aplicação *web*. Entre as principais tecnologias de *frontend* destacam-se Angular, React e Vue.js. A grande maioria dos websites atuais utilizam JavaScript de alguma forma para garantir interatividade e funcionalidades nas páginas *web*. As páginas *web* tradicionais são aplicações multi-página, ou seja, um ficheiro HTML é carregado sempre que o utilizador navega para outra página do *website*. Este sistema torna a experiência do utilizador menos positiva, uma vez que os tempos de espera são maiores em comparação com as novas aplicações de página única, em que apenas são descarregadas do servidor as partes que são necessárias, reduzindo assim os tempos de espera e melhorando a experiência do utilizador.

6.1.1. DOM

O *Document Object Model* (DOM) é uma parte essencial do desenvolvimento de *websites* interativos, uma vez que é uma interface que permite às linguagens de programação manipular o conteúdo, estrutura e estilo de uma página *web*. Funciona em conjunto com documentos HTML ou XML, definindo a lógica dos documentos e a maneira como são acedidos e manipulados (Rascia, 2020).

Um *website* consiste, no seu modo mais básico, num documento HTML. O navegador de internet é um programa que interpreta o código HTML e CSS e apresenta o conteúdo, estilo e estrutura da página que o utilizador vê. Além de analisar o estilo e a estrutura definidos no código HTML e CSS, o browser cria uma representação do documento conhecida como DOM, permitindo ao JavaScript aceder aos elementos do documento como objetos (Rascia, 2020).

Para visualizar o DOM de um *website*, podemos clicar com o botão direito e selecionar a opção “Inspeccionar” ou algo similar (ver Figura 18). Isto abrirá uma nova janela conhecida como “ferramentas de desenvolvimento” com várias opções no cabeçalho. Selecionando a opção “Elementos” podemos visualizar a estrutura do DOM, sendo possível verificar que se assemelha bastante com um documento HTML. Neste caso, cada *tag*¹⁸ é representada como um objeto, havendo a possibilidade de o utilizador expandir cada um deles e ver o que contém, uma vez que o DOM fornece uma representação do documento em árvore¹⁹. O DOM pode ser manipulado utilizando linguagens de script, como JavaScript.



```
<!DOCTYPE html>
<html lang="pt" class="overflow-y-scroll overflow-x-hidden scroll-smooth template-home page-home theme-white site-1" id="id-1">
  <head>
  </head>
  <body class="flex">
    <div hidden id="sprite-plyr">
    </div>
    <!-- Cookies modal -->
    <div class="fixed hidden top-0 left-0 w-screen h-screen bg-black/80 md:p-8 z-[9999] md:backdrop-blur-sm md:place-content-center place-content-end js-cookies-modal">
    </div>
    <div class="fixed top-0 left-0 z-50 flex-col hidden w-screen h-screen p-16 text-white bg-gray-900 js-quickaccess-panel bg-opacity-95 backdrop-filter backdrop-blur-sm">
    </div>
    <header class="fixed flex flex-col w-full transition bg-white">
    </header>
    <main class="lg:mt-16">
    </main>
    <div class="container relative px-0 overflow-hidden shadow-xl xl:rounded-2xl slideshow-container">
      <div class="owl-carousel mt-[50px] md:mt-0 slideshow owl-loaded owl-drag">
        <div class="owl-stage-outer">
          <div class="owl-stage" style="transform: translate3d(-10240px, 0px, 0px); transition: all 0.25s linear 0s; width: 19200px;">
            <div class="owl-item cloned" style="width: 1280px;">
            </div>
            <div class="owl-item" style="width: 1280px;">
              <div data-index="1" data-title="R-Shanghai" class="js-slide bg-no-repeat bg-cover bg-center flex items-center justify-center w-screen lg:w-full" style="height: 1280px; background-image: url('/site/assets/files/1265851/r-shanghai.png');">
                <a class="flex flex-1 w-full h-full" href="https://noticias.uc.pt/artigos/universidade-de-coimbra-entre-as-500-melhores-universidades-do-mundo/" target="blank">
                  <div class="slide-content grid h-full w-full gap-4 text-center text-white text-xl place-content-center">
                  </div>
                </a>
              </div>
            </div>
            <div class="owl-item" style="width: 1280px;">
            </div>
            <div class="owl-item active" style="width: 1280px;">
            </div>
            <div class="owl-item" style="width: 1280px;">
            </div>
            <div class="owl-item" style="width: 1280px;">
            </div>
            <div class="owl-item cloned" style="width: 1280px;">
            </div>
          </div>
        </div>
      </div>
    </div>
  </body>
</html>
```

Figura 18 - Exemplo de informação obtida após clicar no botão para inspeccionar uma página em www.uc.pt.

¹⁸ É um tipo de marcador que delimita secções de um documento HTML. Todas as *tags* encontram-se entre “< >”.

¹⁹ Tipo de estrutura utilizada para organizar dados em hierarquia. Cada elemento da árvore é chamado de nó.

6.1.2. HTML

HTML (*Hypertext Markup Language*) é uma linguagem de marcação utilizada para definir a estrutura de uma página na *web*, sendo um dos pilares fundamentais de qualquer página *web* (Chris K., 2021). É uma linguagem que utiliza *tags* para permitir ao navegador identificar a estrutura desejada pelo autor, ou seja, como a página deve ser apresentada. Foi criada por Tim-Berners-Lee em 1993, após ter proposto um sistema de hipertexto para partilhar toda a informação armazenada em computadores entre físicos do CERN, dando assim origem à *World Wide Web*, juntamente com Robert Cailliau, entre outros (McGrath, 2020). Ao longo dos anos 90 tornou-se popular devido ao crescimento da *web*. A última versão do HTML até ao momento é o HTML5.

6.1.3. CSS

O HTML foi desenvolvido para descrever a semântica²⁰ das componentes do documento *web*, porém não fornece instruções de estilo. Uma vez que o HTML não conseguia responder às necessidades dos programadores para estilizar as páginas, foi introduzido o *Cascading Style Sheets* (CSS), com o objetivo de fornecer um método mais eficiente para a apresentação de páginas *web* em conjunto com o HTML. Foi proposto pela primeira vez em 1994 por Håkon Lie, que trabalhava no CERN juntamente com Tim-Berners-Lee, inventor da *World Wide Web*, porém o primeiro padrão para o CSS foi publicado apenas em 1996 pela W3C (Glass, 2020).

CSS é a linguagem utilizada para definir o estilo das páginas *web*, permitindo definir o aspeto dos diferentes elementos presentes na página, tais como a cor, margens, o tipo e tamanho da letra, e a sua organização, assim como o seu comportamento em diferentes dispositivos com diferentes tamanhos de ecrã. Através da definição de um ficheiro de estilo, é possível obter uma maior flexibilidade e controlo, permitindo que diversas páginas partilhem a mesma formatação e reduzindo assim a repetição de regras em cada página. Uma folha de estilos consiste numa lista de regras, através da utilização de seletores. Um seletor permite selecionar qual ou quais os elementos HTML aos quais se pretende atribuir determinada regra de estilo (Refsnes & W3Schools, 2010).

²⁰ Refere-se à utilização de *tags* HTML para representar o conteúdo de uma página *web*.

6.1.4. JavaScript

JavaScript é a linguagem de programação da *web*. Grande parte dos *websites* modernos utilizam JavaScript, de forma a garantir a interação com o utilizador. Da mesma forma, todos os navegadores modernos incluem interpretadores JavaScript, tornando-se assim a linguagem de programação mais popular da história. É uma linguagem de alto nível, dinâmica, interpretada e não tipada²¹, conveniente para estilos de programação orientados a objetos (Flanagan, 2004). Foi criada pela *Netscape* em parceria com a *Sun Microsystems*, com a finalidade de adicionar interatividade a uma página *web*, sendo lançada inicialmente em 1995 (Silva, 2020).

6.1.5. TypeScript

TypeScript é uma extensão do JavaScript desenvolvida pela *Microsoft*, com o objetivo de ultrapassar as suas limitações, como por exemplo a sua utilização em aplicações de grande escala. Desta forma, TypeScript é um superset do JavaScript, que é uma implementação padrão ECMAScript, e por isso cada programa JavaScript é também um programa TypeScript válido. O compilador TypeScript verifica o código e emite JavaScript, garantindo assim que os programas podem ser executados numa grande diversidade de ambientes (Bierman et al., 2014).

6.1.6. Frameworks

6.1.6.1. Angular

Angular é um *framework* JavaScript que permite aos desenvolvedores criarem aplicações de forma rápida e de fácil manutenção. A principal vantagem na sua utilização está relacionada com a sua boa performance, com a sua arquitetura modular e dinâmica, evitando assim a duplicação de código (Rajendran, 2022). O Angular fornece às aplicações uma estrutura limpa e fácil de compreender e de manter, permitindo a reutilização do código em diferentes aplicações, uma vez que funciona através de módulos (Bielak et al., 2022).

²¹ É uma linguagem de programação em que não é necessário especificar o tipo de dados de determinada variável, permitindo atribuir qualquer valor à mesma.

6.1.6.2. React

React é um *framework* JavaScript criado pelo *Facebook* em 2011 com o objetivo de resolver problemas relacionados com o desenvolvimento de interfaces de utilizador complexas e com conjuntos de dados que são alterados com o tempo. Em 2013 o *Facebook* decidiu tornar o React um projeto de código aberto para incentivar a sua utilização e promover a contribuição da comunidade (Gackenhaimer, 2015). É utilizado para construir interfaces de utilizador interativas e tornar aplicações *web* mais eficientes e com uma redução de código significativa (Herbert D., 2023). Da mesma forma que o Angular, no React também existem componentes reutilizáveis, que quando juntas formam a interface de uma aplicação.

6.1.6.3. Vue.js

Vue.JS é um *framework* JavaScript utilizado para desenvolver a interface do utilizador em aplicações *web*, sendo uma tecnologia relativamente fácil de aprender para qualquer programador. É um dos *frameworks* mais populares e foi desenvolvido de maneira a combinar as melhores características dos restantes *frameworks frontend*, tendo diversas características que contribuem para a sua popularidade, tais como a sua simplicidade e facilidade de utilização, assim como a sua flexibilidade (Hota et al., 2019).

O Vue.js oferece um modelo de programação baseado em componentes, tal como os *frameworks* apresentados anteriormente. No entanto, as suas componentes são conhecidas como Single-File Component²², pois consistem num ficheiro que permite encapsular o *template*, a lógica e a estilização de uma componente.

Para concluir, o Vue.js permite o desenvolvimento de interfaces de utilizador mais eficientes, independentemente da sua simplicidade ou complexidade. Com recursos como o sistema de reatividade e a renderização declarativa do Vue.js e a facilidade de reutilização de componentes, as interfaces de utilizador desenvolvidas com Vue.js oferecem uma experiência fluída e de alto desempenho para os utilizadores²³.

²² Estas componentes possuem apenas um ficheiro com extensão “.vue”.

²³ Adaptado de <https://vuejs.org/guide/introduction.html> (acedido em 4 de junho, 2023)

6.1.6.4. Análise e Comparação de Frameworks

Nesta secção é feita a comparação e avaliação das tecnologias *frontend* descritas acima. Para apresentar esta comparação com maior clareza, são apresentados na Tabela 12 alguns critérios considerados relevantes para a avaliação deste tipo de tecnologias.

Tabela 12 - Critérios de comparação entre os diferentes *frameworks* de *frontend*.

Framework	Angular	React	Vue.js
Linguagem	TypeScript	JavaScript	JavaScript
DOM	Real DOM	Virtual DOM	Virtual DOM
Componentes	Sim	Sim	Sim
Suporte	Google	Facebook	Comunidade
Data Binding ²⁴	Bidirecional	Unidirecional	Bidirecional
Curva de Aprendizagem	Acentuada	Ligeira	Ligeira

Quanto à linguagem, o Angular apresenta alguma desvantagem, uma vez que utiliza TypeScript, ao contrário do React e Vue.js que utilizam JavaScript, que é uma linguagem muito mais popular. O TypeScript é baseado no JavaScript, porém pode ser um aspeto negativo na escolha de um *framework frontend* para alguém que já tenha experiência no desenvolvimento de aplicações *web*, uma vez que estará mais familiarizado com JavaScript, não sendo necessário despende tempo no processo de aprendizagem.

Relativamente ao DOM, o Angular surge mais uma vez diferenciado das restantes alternativas aqui discutidas, uma vez que utilizar Real DOM, o que significa que pode ser menos eficiente no caso de atualizações frequentes da interface do utilizador, uma vez que faz alterações diretamente no DOM e toda a página é novamente renderizada. Por outro lado, o React e Vue.js utilizam Virtual DOM, o que significa que as alterações são feitas numa representação do DOM que é mantida em memória, e só posteriormente é calculada a diferença entre o Virtual DOM e Real DOM, permitindo aplicar as alterações necessárias no Real DOM, contribuindo para atualizações do DOM mais eficientes e com melhor desempenho. Quanto à aprendizagem, o

²⁴ Ligação de dados utilizada nos *frameworks frontend* para estabelecer uma conexão entre a interface de utilizador e a lógica de negócio (código que controla o comportamento da aplicação).

Angular apresenta uma curva de aprendizagem mais acentuada, devido à sua complexidade, à utilização de TypeScript e de conceitos como injeção de dependência ou *decorators* (Saks, 2019; Uzayr et al., 2019).

Contudo, e apesar de o Angular não parecer a melhor opção para a criação da aplicação de cliente do sistema gWater, este foi o *framework* escolhido, uma vez que o mestrando adquiriu conhecimentos e experiência com este **framework** ao longo do primeiro ano do mestrado em Tecnologias de Informação Geográfica, permitindo beneficiar assim de aspetos positivos do Angular, uma vez que é mais adequado para aplicações complexas.

6.1.7. NgRx

O NgRx é um *framework* que tem como finalidade a construção de aplicações reativas em Angular, fornecendo a gestão do estado da aplicação e permitindo o uso de ações para alterar esse estado. O NgRx é inspirado no padrão Redux, que é composto por três conceitos principais: *Store*, *Action* e *Reducer*. A *Store* é responsável por armazenar o estado da aplicação, enquanto as *Actions* são responsáveis por todas as alterações desse mesmo estado, sendo que sempre que uma *Action* é executada, existe um *Reducer* que fica à escuta dessa mesma *Action* para alterar o estado na *Store* (Ekpot W., 2021). Na Figura 19 está representado todo o ciclo de funcionamento do NgRx baseado na lógica do Redux.

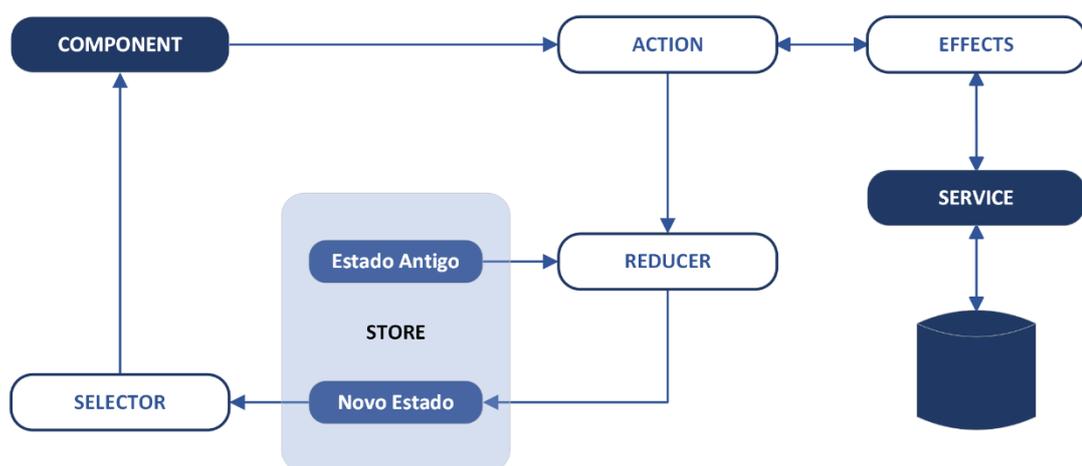


Figura 19 - Ciclo de funcionamento do NgRx

Sempre que uma *Action* é executada, é reconhecida pelo *Reducer*, que é um método que recebe o estado atual e a ação como parâmetros e devolve um novo estado. As ações podem ou não enviar dados para o *Reducer*. Quando o *Reducer* devolve o novo estado, é possível visualizá-lo na componente uma vez que esta subscreve-o através de um *Selector*. As ações podem ainda servir para executar *Effects*, que não são mais que efeitos colaterais que são utilizados sempre que é necessário obter dados que existem fora da aplicação *web*. Sempre que um *Effect* é executado, ele devolve uma ação de sucesso ou de fracasso, que é também reconhecida pelo *Reducer* e que faz prosseguir toda a lógica do NgRx como descrita anteriormente. É ainda importante referir que o estado nunca é alterado diretamente. Em vez disso, o *Reducer* cria sempre uma cópia do estado com as alterações necessárias. Chama-se a isso imutabilidade.

6.2. Backend

As tecnologias de *backend* garantem o bom funcionamento de toda a lógica de uma aplicação *web*. Nesta secção, é apresentada a tecnologia utilizada para a construção da API e o servidor de mapas utilizado. É também apresentada uma justificação para a escolha da tecnologia utilizada para a construção da API em detrimento das restantes alternativas.

6.2.1. Django e Django REST Framework

O Django é um *framework web* baseado em Python, muito popular devido à sua capacidade para resolver questões complexas relativas à criação de um website, tais como a autenticação, conexão com a base de dados, entre outras. Possui também uma interface de administração que permite visualizar informação e executar diversas operações de manipulação de dados existentes na base de dados. Além disso, existem diversos pacotes que adicionam funcionalidades ao Django, como o Django REST Framework, uma vez que possibilita a criação de poderosas APIs *web* e está muito bem documentado (Vincent, 2022).

Tal como já foi referido, o Django simplifica a relação do sistema com a base de dados, uma vez que a torna fácil e de rápido desenvolvimento, sendo baseada em ORM (Object Relational Mapping), que tem como objetivo o envio de dados entre a base de dados e os modelos da aplicação, uma vez que mapeia a relação entre uma tabela da base de dados e um modelo. Desta forma, um atributo de um classe/objeto representa um campo na tabela. A principais vantagens

desta abordagem são a eliminação da necessidade de escrever código SQL e tornar o processo de desenvolvimento mais rápido e livre de erros (Richardson & Ruby, 2008).

A escolha do Django deve-se ao facto de permitir a integração com outras bibliotecas Python que podem ser utilizadas para lidar com dados geoespaciais, uma vez que é baseado em Python. Outras alternativas que implicam a utilização de linguagens de programação diferentes não têm tantas bibliotecas para processar este tipo de dados.

A título de exemplo de bibliotecas que o Django permite utilizar temos o GDAL, o GRASS, o GeoPandas e o NumPy, que permitem a leitura e escrita de formatos de dados geoespaciais.

6.2.2. GeoServer

O GeoServer é um servidor de código aberto escrito em Java que possibilita a partilha, processamento e edição de dados geoespaciais, sendo amplamente utilizado em sistemas de informação geográfica na *web*. Foi projetado com vista à interoperabilidade, o que significa que pode publicar dados de qualquer grande fonte de dados espaciais através de padrões abertos. O GeoServer utiliza os padrões abertos definidos pelo OGC, tais como o *Web Map Service* e o *Web Feature Service*, já abordados anteriormente neste documento. Isto permite ao GeoServer conectar-se a mapas baseados na *web*, tais como OpenLayers ou Leaflet. As fontes de dados podem fornecer dados em diversos formatos, incluindo GeoTIFF, GeoJSON, KML, Shapefile, PostGIS, MySQL, entre outros. O GeoServer oferece ainda recursos de segurança através de autenticação e autorização de forma a proteger os dados geoespaciais, controlando o acesso aos mesmos, assim como a possibilidade de extensibilidade, isto é, adição de novas funcionalidades personalizadas através de plugins e extensões (Iacovella, 2014, 2017).

Para terminar, e em forma de resumo, o GeoServer é uma ótima opção para a partilha de dados geoespaciais na *web* de forma eficiente, eficaz, flexível e interoperável, seja para fins de visualização ou análise. Além disso, é um *software* de código aberto e que pode ser personalizado de acordo com as necessidades de cada organização.

6.3. Protocolo MQTT

MQTT é um protocolo de mensagens leve baseado num sistema de publicação/subscrição de determinado tópico, ou seja, permitindo uma ligação contínua, ao contrário do protocolo HTTP. Este protocolo é normalmente utilizado em aplicações de Internet das Coisas (IoT), quando é necessário haver comunicação entre dois ou mais dispositivos (Hillar, 2017). No contexto do sistema gWater, o protocolo MQTT será utilizado para fazer a comunicação entre a aplicação de cliente e os *scripts* responsáveis pela execução das ferramentas do sistema, de maneira a garantir que o utilizador recebe o estado atual da execução de uma determinada ferramenta sempre que esta passa para uma nova fase, sem pedidos adicionais à API, sendo este o motivo da utilização deste protocolo de comunicação.

7. Implementação

Neste capítulo, será descrito todo o processo de implementação das funcionalidades do sistema gWater, detalhando assim as atividades desempenhadas ao longo deste estágio, assim como as dificuldades encontradas e as soluções utilizadas para as ultrapassar.

7.1. Scripts de Recolha e Tratamento dos Dados

De forma a obter dados de forma contínua e automática, é necessária a implementação de *software* capaz de interagir com as APIs das diversas fontes de dados utilizadas neste projeto. Além disso, os dados recolhidos devem ser sempre tratados, passando por um conjunto de operações para que fiquem em conformidade com os requisitos do sistema e para ser possível visualizá-los na aplicação *web*. O *software* implementado para esta finalidade encontra-se na forma de *scripts* Python, que são arquivos que contêm uma sequência de instruções com o objetivo de realizar determinada operação. Os *scripts* responsáveis pela comunicação com essas APIs e que executam o respetivo *download*, não foram desenvolvidos pelo mestrando, tendo o seu contributo sido feito nos restantes *scripts* de tratamento dos dados. Apesar dos dados recolhidos já terem sido sujeitos a processamento, é necessário fazer o tratamento necessário de modo a satisfazerem os requisitos do sistema gWater e o padrão dos dados utilizados.

Estes processos de recolha e tratamento dos dados foram divididos de acordo com a fonte utilizada, sendo que nesta fase apenas existem dados do CMEMS e dados Sentinel-2. De seguida, será feita uma descrição detalhada dos passos necessários para preparar os dados para a sua aplicação final no sistema, que será tanto para visualização como para utilização em processos de deteção de ocorrências de concentração de clorofila, *inputs* na utilização de ferramentas de geoprocessamento e *inputs* para a construção de gráficos auxiliares para fornecer informação adicional e compreensão dos dados e a sua relação com outros dados.

7.1.1. Dados do CMEMS

Os processos de recolha e tratamento dos dados do CMEMS, tal como indicado na Figura 20, começam com o *script* de *download* dos dados. Este *script* é responsável por fazer uma busca por novos dados de acordo com um conjunto de critérios definidos (ex: área de interesse), através da comunicação com a API existente para esse fim do programa Copernicus. Caso existam novos dados que estejam de acordo com os critérios de busca, serão descarregados para o computador responsável pela execução destes *scripts*.

Uma vez descarregados, os dados passam pela fase de pré-processamento, que tem como principais funções fazer a conversão dos dados para o formato necessário, criar uma imagem de banda única e fazer uma reamostragem dos dados de forma a adequá-los aos requisitos do sistema.

De seguida, são criados os estilos na base de dados, que serão utilizados também para construir a legenda das respetivas imagens. Neste *script*, são calculados os valores mínimo e máximo de cada intervalo, sendo também atribuída uma cor a cada um deles, de acordo com uma das rampas de cores existentes na base de dados. Estes intervalos podem ser calculados através de três métodos diferentes. São eles o método dos intervalos iguais, quantis e JENKS (Arlinghaus et al., 2023), sendo este último o método definido como padrão e que será utilizado para gerar o estilo inicial de todos os dados do CMEMS.



Figura 20 - Figura representativa dos passos/scripts de recolha e tratamento dos dados do CMEMS.

A Figura 21, mostra de forma simplificada o fluxo de processos dentro deste *script*, sendo que os valores mínimos são guardados como uma lista de valores mínimos, assim como os valores máximos e as cores, para que posteriormente sejam utilizados na construção dos ficheiros de estilo que serão carregados no servidor de dados geoespaciais (GeoServer).

O quarto *script* tem como único objetivo o envio das imagens para o servidor.

No *script* seguinte, é feita a reclassificação da imagem, que pode ser feita utilizando o GDAL ou GRASS. Esta reclassificação é feita tendo em conta os intervalos calculados no terceiro *script* e tem como objetivo a simplificação dos dados, atribuindo um valor único para cada um desses intervalos. Por fim, o ficheiro resultante desta operação é enviado para o servidor de maneira a estar disponível para utilização pela aplicação.

O *script* número seis, cria uma *store* no GeoServer e faz a publicação da respetiva *layer*, ficando esta disponível a partir de agora no servidor de mapas.

No penúltimo passo desta sequência, é criado o ficheiro de estilo (sld) baseado no número de intervalos definidos anteriormente e tendo em conta a reclassificação, uma vez que a imagem adicionada ao GeoServer é a imagem reclassificada. É também criado um estilo no GeoServer e associado o respetivo ficheiro de estilo que contém as regras necessárias.

Por último, são eliminados todos os ficheiros desnecessários ao funcionamento do sistema, gerados ao longo desta sequência de processos.

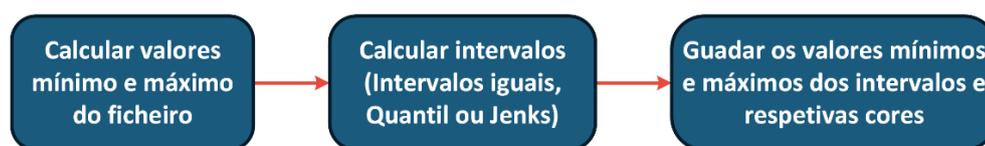


Figura 21 - Esquema representativo do fluxo de processos do *script* responsável por calcular os intervalos necessários para a construção do estilo e da legenda.

Ao todo, nos *scripts* de recolha e tratamento dos dados do CMEMS foram criadas **1333** linhas de código, sem contar com comentários e definições Python importadas de outros ficheiros existentes noutras partes da estrutura do projeto.

7.1.2. Sentinel-2

Os processos de recolha e tratamento dos dados do Sentinel-2 foram divididos em duas fases, de forma a facilitar a compreensão (ver Figura 22). Na primeira fase estão incluídos os *scripts* utilizados para fazer a recolha dos dados, enquanto na segunda fase estão todos os *scripts* utilizados para fazer o respetivo tratamento, assim como a execução de processos necessários para a sua visualização no sistema gWater.

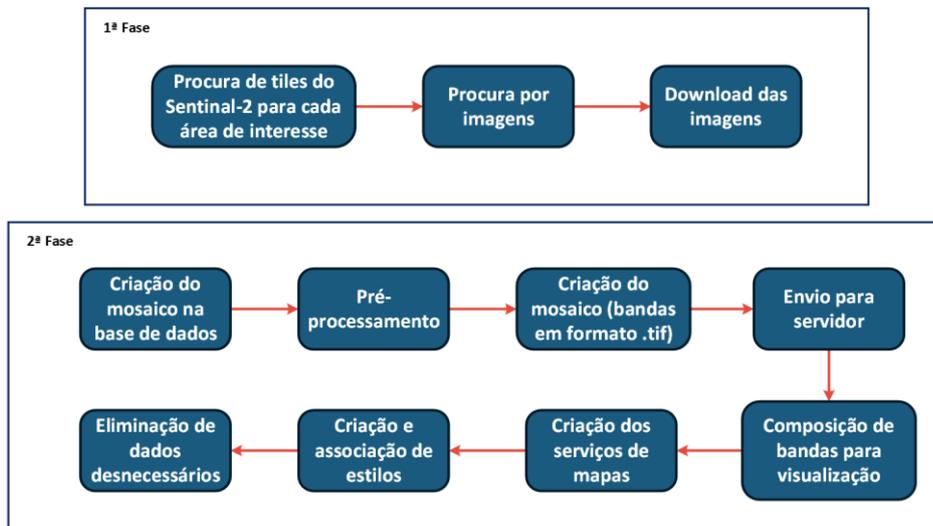


Figura 22 - Figura representativa dos passos/scripts de recolha e tratamento dos dados Sentinel-2.

Tal como no processo de obtenção de dados do CMEMS, os dados do Sentinel-2 são obtidos de forma automática através da comunicação de *scripts* com a API responsável pelo fornecimento destes dados. Estes *scripts* encontram-se na primeira fase, onde existe um *script* para fazer a procura por tiles do Sentinel-2 que cubram as áreas de interesse desejadas, outro para procurar as imagens correspondentes e um terceiro para fazer o seu download.

Relativamente à segunda fase desta sequência de processos, esta inicia-se com a criação de um registo de mosaico na base de dados. Este *script* faz ainda a criação de pastas necessárias para os passos seguintes, e move as imagens para essas pastas.

No *script* seguinte, é feito o pré-processamento dos dados, onde são executados processos, tais como a definição do sistema de coordenadas de acordo com o sistema utilizado no sistema gWater, o recorte da imagem de acordo com uma máscara de água, ou seja, uma vez que só interessa a área correspondente à água, toda a área restante da imagem ficará sem valores associados. É ainda feita uma reamostragem das imagens para ficarem com o mesmo *cellsize*.

Na fase seguinte, para cada registo de mosaico criado anteriormente na base de dados, é criado o mosaico correspondente, que consiste num conjunto de ficheiros *tif* que representam cada uma das bandas da imagem Sentinel-2.

O quarto *script* tem como único objetivo o envio de todos os ficheiros correspondentes às bandas da imagem, para o servidor.

No *script* número cinco, é criado um ficheiro *raster* composto, ou seja, é criado um ficheiro único com alguns dos ficheiros correspondentes a cada uma das bandas obtidos anteriormente. Neste caso, são utilizadas as bandas 4, 3, 2 e 8, que correspondem respetivamente ao “Red”, “Green”,

“Blue” e “NIR”. Posteriormente, serão criadas duas composições de bandas padrão utilizando estas quatro bandas, para que o utilizador possa selecionar qual delas pretende visualizar. As composições padrão serão abordadas de seguida no sétimo *script*, uma vez que correspondem à criação de estilos no GeoServer.

Relativamente ao sexto *script*, este adiciona o ficheiro *raster* composto ao GeoServer, através da criação de uma *store* e de seguida da publicação da *layer*, tal como acontece nos *scripts* relacionados com os dados do CMEMS.

Quanto ao penúltimo *script*, e mais uma vez de igual forma ao que acontece nos *scripts* relativos aos dados do CMEMS, são criados os estilos e adicionados ao GeoServer.

Tal como ilustrado na Figura 23, a sequência de processos para a criação dos estilos tem início com a definição de duas composições de bandas padrão. Estas duas composições padrão são compostas da seguinte forma:

- **Cor Natural:** Canal *Red*: banda 4, Canal *Green*: banda 3, Canal *Blue*: banda 2
- **Falsa Cor:** Canal *Red*: banda 8, Canal *Green*: banda 4, Canal *Blue*: banda 3



Figura 23 – Esquema representativo do fluxo de processos com vista à criação de estilos para as imagens do Sentinel-2.

Como foi referido anteriormente, o ficheiro composto foi criado com as bandas 4, 3, 2 e 8, precisamente nesta ordem. Por isso, nos ficheiros de estilo criados para as duas combinações de bandas referidas acima, a banda 4 corresponde ao número 1, a banda 3 corresponde ao número 2, a banda 2 corresponde ao número 3 e a banda 8 corresponde ao número 4 (ver Figura 24).

```
combs = [  
  # True Color  
  {"RED": "1", "GREEN": "2", "BLUE": "3"},  
  # NIR Color  
  {"RED": "4", "GREEN": "1", "BLUE": "2"}  
]
```

Figura 24 - Combinações de bandas para as imagens Sentinel-2.

Neste caso, estas combinações têm como objetivo a obtenção de uma imagem com cor natural e outra em falsa cor infravermelho com a banda do infravermelho no canal do vermelho. De seguida, são criados os estilos no GeoServer e associados à respetiva *layer*, sendo que apenas no passo seguinte serão criados os ficheiros de estilo com as regras relativas às composições de bandas. Neste caso, são criados dois ficheiros, uma vez que foram definidas duas composições padrão, como já foi referido acima. Por último, os ficheiros são associados aos respetivos estilos criados no GeoServer, que por sua vez estão associados à *layer* correspondente que foi adicionada no *script* anterior.

Por fim, e mais uma vez à semelhança com o que acontece com os dados do CMEMS, são eliminados todos os ficheiros gerados ao longo destes processos e que não são necessários para o correto funcionamento do sistema.

No conjunto de todos os processos descritos nesta secção, foram criadas **1551** linhas de código.

7.2. Desenvolvimento API

No decorrer da implementação do sistema gWater, foram criados diversos *endpoints* com o objetivo de permitir a comunicação da aplicação *web* e de outros elementos do sistema com a API. Todas as operações descritas no ponto anterior, relacionadas com a recolha e tratamento dos dados, exigem comunicação com a API, de maneira a adicionar registos na base de dados ou fazer atualização do estado de cada registo, que é o que permite controlar a sequência prevista de funcionamento dos *scripts*. De igual forma, é necessário fazer pedidos à API para obter registos com um determinado estado, algo que é feito em todos os *scripts* após o primeiro passo. Ainda relativamente aos *scripts* de recolha e tratamento dos dados, todos os estilos, dados de legenda, entre outros, que são adicionados à base de dados, exigem comunicação com a API. Todas estas comunicações com a API exigem a criação de *endpoints*, que são localizações específicas dentro da API que recebem pedidos e devolvem uma resposta. Cada *endpoint* está associado a uma função específica na API, sendo que a combinação do URL com o método HTTP utilizado para fazer o pedido, definem o *endpoint*. Esta comunicação já foi descrita no segundo capítulo.

Para a implementação desta API através do *framework* Django, foi necessário recorrer a diversas aplicações externas que podem ser integradas no nosso projeto, facilitando assim algumas tarefas, uma vez que são aplicações reutilizáveis que se encontram disponíveis para utilização. A

aplicação externa mais importante no contexto desta API é a aplicação Django REST Framework, uma vez que serve de base para o desenvolvimento dos *endpoints*, lógica central do funcionamento de qualquer API. Além disso, o Django REST Framework permite a serialização de objetos e também suporta pacotes como o OAuth2, que é um protocolo que permite proteger os *endpoints* de utilizadores não credenciados, através de um sistema de autenticação e autorização.

Foram seguidas um conjunto de regras de boas práticas que são recomendadas no desenvolvimento de qualquer API, tais como o a utilização do formato JSON, que devido à sua legibilidade e facilidade de manipulação aliado ao facto de ser um formato *lightweight*, se destaca como o formato mais utilizado para esta finalidade. Outra boa prática seguida foi a utilização de *serializers*, que permitem não só validar a estrutura dos dados recebidos durante a criação e atualização dos mesmos, mas também realizar operações CRUD²⁵ à base de dados, tal como acontece no sistema gWater. Além disso, todos os nomes dos *endpoints* não utilizam verbos, uma vez que os métodos HTTP (GET, POST, PUT, DELETE) são suficientes para descrever o tipo de pedido que é feito à API, sendo que os *endpoints* devem representar recursos e não operações a serem realizadas. Usar verbos nos nomes dos *endpoints* pode levar à criação desnecessária de *endpoints* uma vez que haveria um para cada operação que se pretende realizar relativa a determinado recurso, ou então pode induzir em erro ou ser menos intuitiva, seguindo assim contra a convenção REST. Por último, é também considerado boa prática o uso de códigos na resposta HTTP, sendo que neste sistema, além dos códigos de status HTTP que seguem a norma RFC 7231²⁶ definidos pela Internet Engineering Task Force (IETF) e que indicam o resultado da solicitação efetuada, também é devolvido um código específico do sistema gWater.

Na Tabela 13 são apresentados os códigos de status padrão das respostas HTTP tendo em conta as cinco classes existentes, que são definidas pelo primeiro número do código. A primeira classe indica que o servidor reconhece e está a processar a informação. A segunda classe indica que o servidor processou as solicitações com sucesso. A terceira classe indica que o servidor recebeu a solicitação, mas há um redireccionamento para outro lugar. A quarta classe indica que ocorreu um erro do lado do cliente, tal como página não encontrada. A última classe indica que o cliente fez uma solicitação válida, mas ocorreu algum erro no servidor, impedindo-o de concluir a solicitação.

²⁵ CRUD é um acrónimo que representa as operações básicas que se podem fazer à base de dados. Em inglês significa Create, Read, Update e Delete, podendo traduzir-se para português como Criar, Ler, Atualizar e Eliminar.
²⁶ <https://www.rfc-editor.org/rfc/pdf/rfc7231.txt.pdf>

Por outro lado, na Tabela 14 são apresentados exemplos de código de status HTTP padrão em conjunto com códigos específicos no contexto do sistema gWater, assim como a descrição para cada conjunto de códigos.

Tabela 13 - Códigos de status de respostas HTTP

Status HTTP	Significado
1XX	Respostas Informativas
2XX	Respostas bem-sucedidas
3XX	Mensagens de redirecionamento
4XX	Respostas de erro do lado do cliente
5XX	Respostas de erro do lado do servidor

Tabela 14 - Exemplo de códigos de status HTTP associados aos códigos específicos do sistema gWater.

Status HTTP	Código	Descrição
200	S20	Dados devolvidos com sucesso
201	S21	Nova instância criada com sucesso
201	S22	Instância foi atualizada com sucesso
200	S23	Instância foi apagada com sucesso
200	S24	Todas as <i>instâncias</i> foram eliminadas com sucesso
201	A21	Foram criados tokens de acesso com sucesso
201	G21	Nova instância do GeoServer foi criada com sucesso
201	G22	A instância do GeoServer foi atualizada com sucesso
201	G23	A instância do GeoServer foi eliminada com sucesso

Quanto às aplicações desenvolvidas especificamente para este projeto, estas encontram-se divididas de acordo com as componentes apresentadas no diagrama de componentes da API, apresentado no capítulo anterior. De seguida, é feita uma breve descrição de cada uma das aplicações Django desenvolvidas para este projeto, dando especial atenção às componentes relacionadas com a obtenção, manipulação e apresentação de informação geográfica. As tabelas

com os *endpoints* implementados encontram-se no Apêndice B, contendo a indicação do URL, método HTTP e uma breve descrição do *endpoint*.

7.2.1. Aplicações Django

A aplicação “**authapi**”, que corresponde à componente “Autenticação” representada no diagrama de componentes da API (ver Figura 17), tem como objetivo lidar com todos os processos relacionados com a autenticação dos utilizadores no sistema gWater. Esta aplicação interage com o pacote OAuth2, que em conjunto com o Django REST Framework permite obter um acesso controlado às diferentes funcionalidades do sistema. Por um lado, o pacote OAuth2 permite autenticar um utilizador, fornecendo um *token* de acesso, enquanto o Django REST Framework controla quem tem acesso a diferentes partes da API, através de diferentes classes de permissão. Além das funções associadas com a autenticação dos utilizadores, esta aplicação é também responsável pela gestão das áreas de interesse, parte importante da lógica de funcionamento do sistema gWater, uma vez que cada utilizador está associado a uma ou mais áreas de interesse, tendo por isso, acesso apenas a dados relativos a essas áreas.

A aplicação “**charts**”, que corresponde à componente “Gráficos” (ver Figura 17), contém todos os *endpoints* necessários para a criação de gráficos no sistema, que é algo que foi definido na lista de requisitos do sistema. Os *endpoints* criados permitem a criação de histogramas, gráficos de dispersão e gráficos de evolução temporal, através da utilização de definições Python criadas para a construção de cada um destes tipos de gráficos e que utilizam diversas bibliotecas, tais como GDAL, GeoPandas e NumPy, que permitem trabalhar com dados geoespaciais. A aplicação de cliente faz os pedidos HTTP necessários a estes *endpoints* sempre que é necessário obter valores para a criação de um gráfico, fornecendo todos os dados necessários para que o *endpoint* possa efetuar a sua lógica e devolver esses valores na sua resposta, caso esta seja de sucesso. Quando esses valores são devolvidos na resposta da API, a aplicação de cliente constrói o gráfico com o auxílio da biblioteca D3.js, utilizando os respetivos valores fornecidos pela API. De salientar que foi necessário fazer ajustes nas definições Python criadas originalmente para as tornar mais eficientes em termo de utilização de memória, uma vez que o equipamento pessoal utilizado tem algumas limitações a nível do *hardware*, sendo necessário fazer ajustes para jogar com a rapidez de execução e uso de memória. Desta forma, foram aplicadas soluções como o ajuste do tamanho do

bloco de dados que é lido de cada vez, adicionando o parâmetro *block-size* a estas definições Python e permitindo jogar entre estes dois fatores importantes.

Quanto à aplicação “**logs**” (ver Figura 17), esta é responsável por guardar todos os registos de operações e erros que possam ocorrer durante a execução de determinados processos, isto é, operações efetuadas nos *endpoints* das restantes aplicações Django.

A aplicação “**geosrv**”, que está associada à componente “GeoServer” apresentada no diagrama de componentes da API (ver Figura 17), é responsável pela gestão de tudo o que está relacionado com a criação de serviços no GeoServer, o servidor de mapas do sistema gWater. Esta aplicação conta com um conjunto de *endpoints* que permitem operações relacionadas com *workspaces*, *stores*, *layers*, estilos e dois *endpoints* utilizados para obter os WMS e WFS sempre que é necessário visualizar dados que estão presentes no GeoServer. De notar que, tal como já explicado anteriormente, cada aplicação Django tem um conjunto de URLs que permitem identificar a respetiva classe (conhecidas como Class-Based View) que existe nos ficheiros que contêm as *views*, que é onde está toda a lógica executada pelo *endpoint*, e que em conjunto com o método de requisição HTTP (GET, POST, PUT, DELETE) permite executar a *view* pretendida dentro dessa classe, que são definições Python que recebem o pedido HTTP como parâmetro, podendo aceder aos respetivos dados enviados através do mesmo. Por exemplo, no caso das *layers* existem dois URLs definidos nesta aplicação para lidar com as mesmas, e que estão associados a duas classes diferentes, uma para lidar com a criação de *layers* e obtenção de uma lista contendo todas as *layers* existentes e outro para lidar com uma *layer* específica. O primeiro apenas aceita os métodos GET e POST, para requisitar todas as *layers* e para criar uma *layer* respetivamente, contendo por isso duas definições Python. O segundo, contém os métodos GET, PUT e DELETE, que permitem obter a *layer* especificada, alterar a *layer* especificada e apagar a *layer* especificada. Neste caso, a identificação da *layer* é passada no próprio URL utilizado para fazer o pedido à API. Este tipo de URL é chamado de URL dinâmico ou parametrizado, uma vez que contém uma variável, sendo que normalmente está associada ao id de uma instância ou qualquer outro identificador.

Na Figura 25, estão presentes os URLs descritos acima, que são utilizados para executar todas as operações relacionadas com as *layers* no GeoServer, e onde se pode verificar, no caso do segundo URL, que a variável passada é do tipo *string* e é chamada de “lyr”. Na Figura 26, é demonstrado como podemos aceder à variável “lyr” passada no URL dinâmico, uma vez que ela é passada para a *view* correspondente como um argumento.

```

39     # GeoServer Layers
40     path(
41         'layers/', GeoServerLayers.as_view(),
42         name='geoserver-layers'
43     ),
44
45     path(
46         'layer/<str:lyr>', GeoServerLayer.as_view(),
47         name='geoserver-layer'
48     ),

```

Figura 25 - URLs utilizados para executar operações relacionadas com as layers no GeoServer.

```

217 class GeoServerLayer(APIView):
218     """
219     Get, edit, delete GeoServer Layer
220     """
221
222     permission_classes = [
223         permissions.IsAdminUser|IsMatereo,
224         TokenHasReadWriteScope
225     ]
226     parser_classes = [JSONParser]
227
228     @swagger_auto_schema(responses={200: "Data successfully returned"})
229     def get(self, request, lyr):
230         """
231         Method GET - Retrieve a specific layer
232         ---
233         """

```

Figura 26 - View correspondente ao método GET da classe responsável por lidar com uma *layer* específica do GeoServer.

Para executar todas as operações relacionadas com as *layers* no GeoServer, esta aplicação comunica diretamente com a API REST do GeoServer, através de pedidos HTTP com os respetivos métodos associados, tal como a aplicação de cliente faz pedidos a esta API. Resumindo, sempre que o utilizador adiciona uma *layer* ao mapa, a aplicação de cliente faz um pedido à API do sistema gWater que por sua vez faz outro pedido à API do GeoServer. Esta lógica de funcionamento está presente em diferentes esquemas presentes neste relatório, tal como no diagrama de componentes da API (ver Figura 17), onde é visível a ligação da componente “GeoServer” da API ao contentor “Infraestrutura de Dados Espaciais” do sistema gWater. Desta forma, podemos dizer que esta aplicação Django, e por consequência a API, servem de ponte para obtenção dos WMS e WFS por parte da aplicação *web*. Esta abordagem permite não ter de lidar com outro sistema de gestão de utilizadores e com as suas permissões, uma vez que todas as

solicitações ao servidor de mapas passam primeiro pela API, que já faz esse controle. Desta forma, ao isolarmos o GeoServer estamos a reduzir a complexidade de configurações necessárias e a manter a segurança no acesso aos dados geográficos.

Quanto à aplicação “**dset**”, que está associada à componente “Datasets” presente no diagrama de componentes da API (ver Figura 17), está relacionada com todos os processos de recolha e tratamento dos dados do CMEMS e dados Sentinel-2, sendo por isso uma peça bastante importante do sistema gWater. Esta aplicação é responsável pela gestão de todos os *datasets* associados às imagens Sentinel-2, assim como pela gestão das camadas do CMEMS e das respetivas variáveis (temperatura, clorofila, etc...), apresentando-se como a aplicação em todo o projeto Django com um maior número de *endpoints*. Durante os processos executados pelos scripts de recolha de dados do CMEMS e do Sentinel-2, que já foram explicados neste capítulo, são utilizados muitos dos *endpoints* existentes nesta aplicação, uma vez que contém *endpoints* como a criação de uma instância de *layer* CMEMS ou de um *dataset* Sentinel-2. É ainda através dos *endpoints* existentes nesta aplicação que é feita a gestão dos “ABDatasets”, que estão associados aos dados WFS que mostram as áreas com grande concentração de clorofila, indicativo de presença de algas.

A aplicação “**geovis**”, associada à componente “Geovisualização”, apresentada no diagrama de componentes da API (ver Figura 17) é a responsável pela gestão das camadas Sentinel-2, camadas *Algae Bloom* e ainda com os estilos e legendas das camadas do CMEMS. Estes estilos são um conjunto de estilos padrão, cada um com uma determinada paleta de cores, que são utilizados para criar os conjuntos de regras baseados nos intervalos e método utilizados para criar a simbologia de cada camada. Estas regras são depois utilizadas para criar um ficheiro de estilo que será carregado no GeoServer, permitindo obter as simbologias desejadas para cada camada. As regras definidas para cada camada, baseadas nos intervalos e método escolhido, resultam numa instância de legenda que está associada a uma camada específica, que por sua vez pode ter mais do que uma instância de legenda associada. Portanto, os *endpoints* associados aos estilos permitem fazer a gestão dos estilos padrão (paletas de cores) do CMEMS existentes no sistema, enquanto os *endpoints* associados às legendas das camadas CMEMS permitem fazer a gestão das regras definidas para cada legenda.

Por último, a aplicação “**tools**”, associada à componente com o mesmo nome no diagrama de componentes da API (ver Figura 17), é a responsável pela gestão dos dados associados à execução de ferramentas. Todas as ferramentas existentes no sistema estão registadas na base de dados, servindo como referência para executar as definições em Python necessárias para cada ferramenta. Além disso, cada ferramenta tem associados a si um conjunto de parâmetros, que são campos que servem de *input* para o utilizador fornecer informação necessária para a execução de

qualquer ferramenta. Estes *endpoints* servem para fazer a gestão destes registos, entre outros. Quando o utilizador vê uma lista de ferramentas disponível na aplicação *web*, não é mais que informação devolvida pela API como resposta a uma solicitação HTTP, que neste caso é do tipo GET e é feita ao *endpoint* responsável por devolver todos os registos da tabela “*tools*” existente na base de dados. De notar que estes *endpoints* não são responsáveis pelas operações de processamento de informação geográfica, mas sim de gestão de todos os registos associados. Ainda neste capítulo, será explicada toda a lógica implementada para permitir a execução de ferramentas no sistema, onde será apresentada uma figura com as tabelas necessárias e detalhadas todas as relações existentes entre elas.

No conjunto de todas as aplicações Django apresentadas nesta secção, foram implementadas **16 619** linhas de código. Quanto aos *endpoints* criados ao longo deste estágio, e podendo não corresponder ao número de *endpoints* apresentados no Apêndice B, uma vez que alguns dos *endpoints* aí apresentados foram reutilizados de projetos passados, foram criados cerca de 75 *endpoints*.

7.3. Aplicação Web

Tal como já foi mencionado, a aplicação *web* foi construída com recurso ao *framework* Angular, baseado em TypeScript, que é uma extensão do JavaScript, e é indicado para a construção de aplicações *web* dinâmicas e interativas. Uma das principais características do Angular é o facto de ser baseado em componentes, permitindo a construção de aplicações organizadas em módulos, permitindo reutilizar esses módulos em diversas partes da aplicação. Segundo as boas práticas descritas na documentação do Angular, cada componente deve ser responsável por uma funcionalidade, permitindo a sua reutilização. Este comportamento segue o princípio da responsabilidade única, promovendo um código mais organizado e com responsabilidades bem definidas. Por exemplo, todas as chamadas à API feita pela aplicação *web* do sistema gWater são feitas em serviços Angular, que permitem definir funções ou funcionalidades necessárias para o funcionamento da aplicação. Além de permitir que várias componentes utilizem o mesmo serviço para fazer uma chamada à API ou qualquer outra operação que possa ser partilhada por várias componentes, esta abordagem ajuda a dividir a aplicação *web* em várias unidades lógicas que podem ser reutilizadas, que são as componentes.

A implementação da aplicação *web* consistiu na aplicação final destinada aos utilizadores, uma vez que não houve tempo para implementar a interface de utilizador da página de gestão,

destinada à gestão do sistema. A parte central do desenvolvimento desta aplicação foi a página de visualização e manipulação dos dados geográficos, que já foi descrita anteriormente, e que será o foco nesta secção do relatório. De seguida serão apresentadas as principais componentes Angular criadas para a obtenção dessa página, assim como os serviços necessários, e apresentados exemplos práticos implementados na aplicação.

No total, foram criadas **10 770** linhas de código na implementação da aplicação web.

7.3.1. Componentes

A página de visualização dos dados, que é a página principal deste sistema, é composta por diversas componentes, as quais utilizam diversos serviços para diversas finalidades, tais como pedidos à API ou adicionar e remover camadas do mapa. Na Figura 27, é apresentado um esquema em que se demonstra as componentes Angular utilizadas para criar essa página.

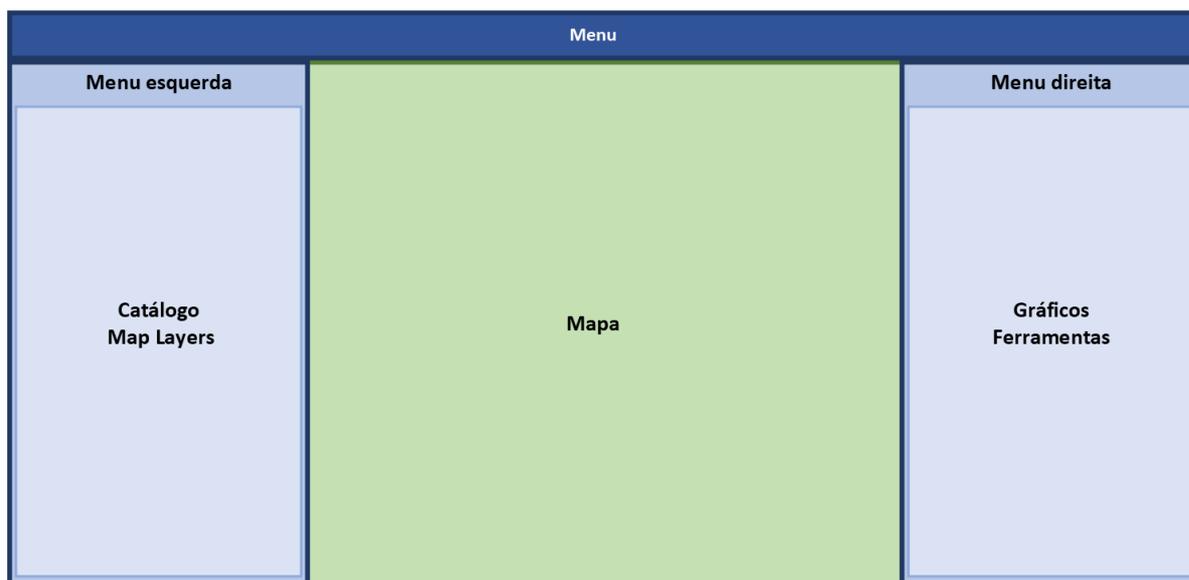


Figura 27 - Esquema representativo da interface da página de visualização e manipulação dos dados.

Tal como demonstrado acima, e ignorando a componente “Menu” que está associada ao menu principal no cabeçalho, existem 7 componentes principais nesta página, que podem conter ainda mais componentes aninhadas dentro de si, chamadas de subcomponentes, como é o caso da componente “charts”, que está associada à visualização dos gráficos, e que possui uma componente para cada tipo de gráfico, consoante a escolha do utilizador. De notar que todas as componentes

aqui apresentadas, à exceção da componente “Menu” são subcomponentes de uma componente principal criada para agrupar todas as componentes desta página, e que por sua vez é uma subcomponente da componente “main”, que contém as restantes componentes desta aplicação.

O elemento central desta página é a componente “Mapa”, que aparece no centro e tem como objetivo apresentar os dados num mapa Leaflet. Para isso, é necessário haver um catálogo com todas as camadas que o utilizador pode adicionar ao chamado “Map Layers”, que é uma subcomponente da componente “Menu esquerda”, e que pretende conter todas as camadas (ativas ou não no mapa) que o utilizador adicionou para visualização ou manipulação, uma vez que apenas as camadas que aparecem no “Map Layers” poderão ser adicionadas ao mapa, utilizadas para a construção de gráficos ou manipuladas através das ferramentas. O catálogo onde o utilizador pode visualizar todos os dados existentes no sistema separados por área de interesse e por variável (no caso das camadas do CMEMS), encontra-se na componente “Catálogo”, que também é uma subcomponente da componente “Menu esquerda”. Para alternar entre as componentes “Catálogo” e “Map Layers”, são utilizados dois botões no topo da componente “Menu esquerda”.

Do lado direito da página existe um segundo menu lateral, que está associado à componente “Menu direita”, contendo duas subcomponentes, a componente “Gráficos”, que tal como já foi referido, possui subcomponentes para cada tipo de gráfico, e a componente “Ferramentas”, que permite ao utilizador escolher entre uma lista de ferramentas e consoante a escolha, introduzir os parâmetros necessários para a sua execução.

7.3.2. Serviços

No contexto da aplicação *web* do sistema gWater, os serviços foram utilizados principalmente para fazer pedidos à API, mas também para definir as funções que permitem adicionar e remover camadas do mapa Leaflet. Desta forma, qualquer componente consegue interagir com o mapa.

Os métodos dos serviços utilizados para fazer os pedidos à API contêm os URLs das “Views” existentes no projeto Django, e através do método HTTP utilizado, direciona o pedido para o *endpoint* pretendido.

Foram implementados diversos serviços na aplicação *web*, em que cada um contém um conjunto de funções, que na sua maioria servem para obter dados do servidor, tal como já foi referido anteriormente. Existe o serviço relacionado com a autenticação de utilizadores, serviço relacionado com o mapa Leaflet, serviço associado às ferramentas e serviço associado à obtenção

de dados para a construção de gráficos. Cada um destes serviços contém um conjunto de métodos para realizar diversas operações específicas, tais como pedir dados para um tipo de gráfico específico. Portanto, cada serviço agrupa um conjunto de métodos associados a uma classe (serviço) e estão relacionados com o mesmo módulo. De seguida, é apresentado na Figura 28 um exemplo de um método definido dentro do serviço “ToolschartsService”, que contém todos os métodos para pedir os dados necessários à API para a construção de todos os tipos de gráficos.

```
9  @Injectable({
10     providedIn: 'root'
11 })
12 export class ToolschartsService {
13
14     constructor(private http: HttpClient) { }
15
16     getHistogramData(token: Token, rst: Layer, nint: number, band: string, amp: number, maxval: number) {
17         var httpOptions = {
18             headers: new HttpHeaders({
19                 'Content-Type': 'application/json',
20                 'Authorization': token.token_type + ' ' + token.access_token
21             }),
22             params: new HttpParams()
23                 .set('rst', JSON.stringify(rst))
24                 .set('nint', nint.toString())
25                 .set('band', band)
26                 .set('amp', amp)
27                 .set('maxval', maxval)
28         };
29
30         return this.http.get<HistogramApi>(api.histogram, httpOptions);
31     };
32 }
```

Figura 28 - Método getHistogramData do serviço ToolschartsService

Este método é utilizado para obter os dados necessários para a construção de um histograma. Para isso, é importante ter em conta que este método serve para as camadas CMEMS e Sentinel-2, por isso possui um conjunto de argumentos que permitem passar os dados necessários quando são chamados. De notar que alguns destes argumentos não recebem valor consoante o tipo de camada escolhida para a construção do gráfico, como é o caso do argumento “band”, que apenas faz sentido caso seja uma camada do Sentinel-2. O *endpoint* responsável pelo tratamento deste pedido consegue identificar qual o tipo de camada que está a ser tratada e, assim, optar por uma de duas definições Python que permitem extrair os dados dos ficheiros das imagens, utilizando diversas bibliotecas Python capazes de lidar com dados geoespaciais, como o GDAL ou GeoPandas.

7.3.3. Gestão do Estado da Aplicação

Para fazer a gestão do estado da aplicação, foi utilizada a biblioteca NgRx, inspirada no Redux, que permite a partilha de informação entre componentes de forma mais simples, sem a necessidade de utilizar a comunicação entre componente pai e componente filho. A sua lógica de funcionamento já foi explicada na secção 6.1.7, quando foi feita a apresentação das tecnologias utilizadas no projeto gWater. Nesta secção, será explicada de que forma foi utilizada esta biblioteca e apresentados exemplos práticos da sua utilização.

Como já foi referido, os conceitos principais da lógica do NgRx são a *Store*, as *Actions*, os *Reducers*, os *Selectors* e os *Effects*. No contexto do sistema gWater, foram utilizadas *Actions* para operações como adicionar uma camada do catálogo ao Map Layers, que adiciona um elemento à propriedade *layers* do MapState, que é um dos subestados do estado principal, e que contém propriedades relativas ao mapa. Outra operação em que é utilizada uma *Action*, é na funcionalidade de mostrar/ocultar a camada no mapa, através de uma *checkbox*, sendo que a ação que é disparada é sempre a mesma, alterando uma propriedade do objeto do tipo *Layer* (interface) correspondente. Essa alteração é feita através de um *Reducer*, que altera a propriedade “*active*” de um objeto “*layer*” que está dentro do array “*layers*”, que é uma propriedade do subestado “*MapState*”. A Figura 29 apresenta a implementação deste *Reducer*, demonstrando a lógica descrita acima.

```
90 // Activate/Deactivate Layer
91 on(mapAction.ToggleLayer, (state, { payload }) => {
92   const updatedLayers = state.layers.map(layer => {
93     if (layer.id === payload.id) {
94       return { ...layer, active: !layer.active };
95     }
96     return layer;
97   });
98
99   return { ...state, layers: updatedLayers };
100 },
```

Figura 29 - *Reducer* NgRx que altera a propriedade "active" de um objeto "Layer".

Neste exemplo, é devolvido um novo objeto de estado com o *array* das *layers* atualizado, seguindo assim o princípio de imutabilidade do estado, que é uma prática recomendada no desenvolvimento com NgRx.

7.4. Execução de Ferramentas

7.4.1. Lógica de Funcionamento com Protocolo MQTT

A execução de ferramentas no contexto do sistema gWater pode levar a tempos de espera demasiado longos, uma vez que as operações a executar são complexas e os dados têm um tamanho considerável, prejudicando severamente a experiência do utilizador. Por esse motivo, foram utilizadas técnicas que permitem evitar tempos de espera, permitindo ao utilizador continuar a utilizar o sistema enquanto o *output* da execução da ferramenta é gerado. Para isso, é necessário que o sistema notifique o utilizador aquando da conclusão dos processos de execução da ferramenta, assim como do processo em que se encontra essa execução, ou seja, apresentar continuamente o progresso ao longo de todos os processos. Para isso, foi utilizado o protocolo MQTT, uma vez que permite uma ligação contínua, sendo baseado num sistema de publicação/subscrição.

Tal como mostra a Figura 30, é necessário haver uma comunicação contínua entre a aplicação cliente (representada na figura por “APP Cliente”) e os *scripts* responsáveis pelos processos necessários para a execução de uma ferramenta (representados na figura por “gWater Toolbox”). Para isso, é necessário um broker MQTT, que atua como um intermediário neste processo de comunicação entre duas entidades, gerindo as publicações e subscrições. Um broker MQTT recebe publicações, filtra essas publicações pelo tópico e, de seguida, distribui essas mensagens para os subscritores do mesmo tópico. Seguindo esta lógica de funcionamento, os *scripts* de execução de ferramentas publicam num determinado tópico, ao qual a aplicação cliente subscreve previamente aquando do início da execução da ferramenta, possibilitando a comunicação entre estes dois elementos. Este broker MQTT encontra-se em execução num contentor criado especificamente para esse fim, tal como já foi referido anteriormente na apresentação da arquitetura do sistema.

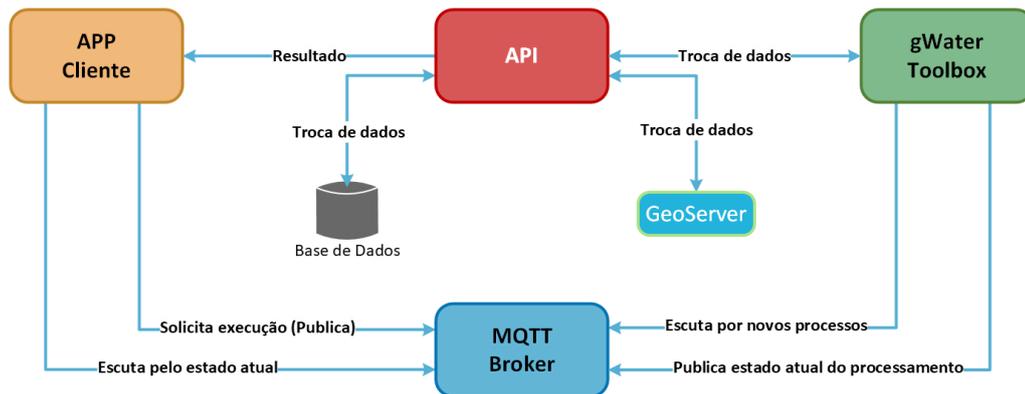


Figura 30 - Esquema demonstrativo da lógica de comunicação entre a aplicação de cliente e os *scripts* de execução de ferramentas, através do protocolo MQTT.

7.4.2. Scripts do Módulo gWater Toolbox

A Figura 31 apresenta um esquema representativo do fluxo de funcionamento dos *scripts* responsáveis pela execução de ferramentas, sendo que cada etapa neste fluxo pode não corresponder a um *script* individual.

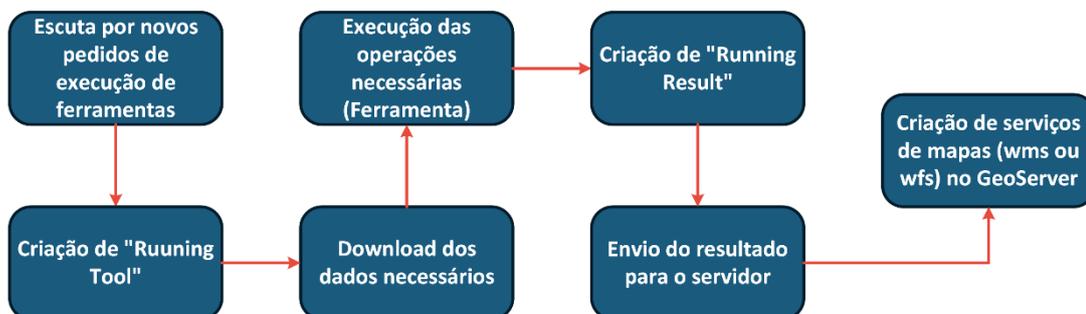


Figura 31 - Esquema representativo do fluxo de funcionamento dos *scripts* responsáveis pela execução de ferramentas no sistema gWater.

Sempre que um utilizador dá início à execução de uma ferramenta, é adicionado um registo a uma tabela "RunningTool" existente na base de dados que contém todas as execuções de ferramentas realizadas no sistema. Nesse registo, são armazenadas informações essenciais como o identificador único dessa "running tool", o utilizador que a criou, a ferramenta associada à mesma,

entre outras informações. Na Figura 32, é demonstrado o relacionamento entre algumas das tabelas criadas para assegurar a execução de ferramentas.

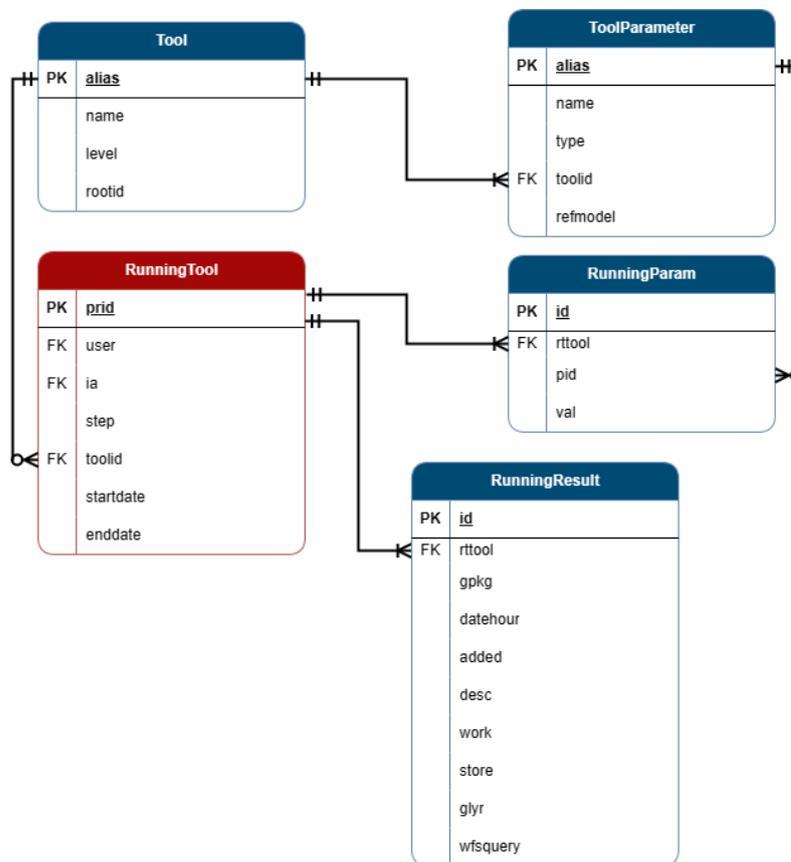


Figura 32 - Diagrama de Entidade e Relacionamento de parte das tabelas criadas para assegurar a execução de ferramentas no sistema gWater.

Todas as ferramentas existentes no sistema correspondem a um registo na tabela “Tool”, que terá pelos menos um parâmetro associado, parâmetros estes presentes na tabela “ToolParameter”. Cada ferramenta tem um ou mais parâmetros, e por isso, a relação no sentido ferramenta parâmetro é de um registo da tabela “Tool” para um ou muitos registos da tabela “ToolParam”, porém, no sentido contrário, será de um para um, uma vez que cada parâmetro está associado a apenas uma ferramenta, não havendo parâmetros partilhados por várias ferramentas. Da mesma forma, funciona a relação entre as tabelas “RunningTool” e “RunningToolParam”. No entanto, os dois pares de tabelas encontram-se ligados, uma vez que para cada registo da tabela “Tool” poderá existir zero ou muitos registos da tabela “RunningTool”, uma vez que cada “Running Tool” tem de utilizar uma ferramenta (registo) da tabela “Tool”. Seguindo a mesma lógica, para cada registo da tabela “ToolParam” poderá existir zero ou muitos registos da tabela

“RunningParam”, que são os parâmetros utilizados por cada “Running Tool”. Por fim, a tabela “RunningResult” serve para guardar os dados associados aos resultados de cada “Running Tool”, tais como o nome e área de interesse do ficheiro gerado, a ferramenta utilizada, entre outros. De salientar que cada “Running Tool” pode gerar múltiplos ficheiros, e por isso existe uma relação de um ou muitos nesse sentido, sendo que no sentido contrário, para cada resultado só poderá existir uma “Running Tool” associada.

Voltando ao esquema representativo do fluxo de funcionamento dos *scripts* responsáveis pela execução de ferramentas, verificamos que existe um *script* que está à escuta num determinado tópico, aguardando por uma publicação da aplicação cliente que dê início a todo o processo. Ainda no mesmo *script*, sempre que uma mensagem é recebida e tratada com sucesso, é feito um pedido à API por parte do *script* para adicionar um novo registo na tabela “Running Tool”, uma vez que os dados necessários para fazer esse pedido foram fornecidos na mensagem via MQTT. Esta lógica está demonstrada na Figura 30, onde se pode ver que a aplicação cliente faz uma solicitação de execução (publicação num determinado tópico) e, por outro lado, o contentor gWater Toolbox fica à escuta por novas solicitações (publicações no tópico subscrito). É também visível a comunicação do contentor “gWater Toolbox” com a API, de maneira a adicionar a “Running Tool”. Essa comunicação é feita através do protocolo HTTP. Além disso, a API comunica com a base de dados para adicionar o registo pretendido.

O processo seguinte, é o *download* dos dados, uma vez que estes se encontram no servidor, porém é necessário obter esses dados para efetuar as operações necessárias no computador responsável pelo controlo do sistema. Mais uma vez, o contentor “gWater Toolbox” comunica com API para obter os ficheiros desejados.

No processo seguinte são executadas as operações relacionadas com o processamento de dados geográficos, ou seja, é onde se faz o processamento relacionado com a ferramenta escolhida pelo utilizador. Neste momento, e apesar de haver várias ferramentas registadas na tabela “Tools”, apenas uma funciona, que é a ferramenta capaz de identificar valores elevados de clorofila através de *datasets* do Sentinel-2, podendo indicar a presença de algas. Neste sentido, o *script* responsável por fazer este processamento relacionado com a ferramenta escolhida pelo utilizador, foi preparado apenas para lidar com esta ferramenta específica, uma vez que ocorreram diversas limitações, sendo a mais impactante o tempo disponível para a realização destes processos. Para o sistema conseguir lidar com todas as ferramentas pretendidas, será necessário fazer alterações significativas neste *script*, assim como a implementação de novas definições Python na gWater Toolbox com a utilização de bibliotecas como o GDAL, GRASS, GeoPandas entre outras.

De seguida, é criado um registo na tabela “Running Results”, com os dados associados aos resultados obtidos após a execução da ferramenta. Este processo ocorre no mesmo *script* que executa a ferramenta, apesar de estar representado na Figura 31 como sendo diferentes etapas do processo geral.

O *script* seguinte tem como objetivo o envio dos dados para o servidor, que no caso da única ferramenta disponível no sistema é o envio das geometrias para uma tabela associada à tabela “Running Results” que guarda geometrias associadas aos resultados. Neste *script* do contentor “gWater Toolbox” volta a ser feita uma comunicação com a API para enviar os respetivos resultados.

Por último, são criados os serviços de mapas associados aos resultados, podendo ser do tipo WMS ou WFS. No caso da ferramenta de identificação de algas através de dados do Sentinel-2, são criados WFS com os polígonos representativos das áreas com ocorrências de algas. Para isso, o contentor “gWater Toolbox” comunica com a API, que, por sua vez, comunica com o GeoServer, tal como respresentado na Figura 30, permitindo assim a visualização dos resultados obtidos. Caso os resultados obtidos sejam do tipo WMS, será necessário proceder à criação dos respetivos estilos das imagens, à semelhança do que acontece nos *scripts* de recolha de dados.

De notar que, em cada uma das etapas descritas acima, o *script* responsável por essa etapa faz uma publicação num determinado tópico que a aplicação de cliente subscreveu assim que deu início à execução da ferramenta. Essas publicações contêm mensagens indicativas do processo no qual se encontra a execução da ferramenta. Desta forma, o utilizador está a par do progresso destas operações, podendo continuar a utilizar o sistema normalmente. Após a finalização de todos os processos, o resultado ou resultados da ferramenta executada são adicionados ao catálogo de dados do sistema, sendo que o utilizador pode adicionar esses dados ao Map Layers para visualização dos mesmos.

Na implementação dos *scripts* descritos nesta secção, foram criadas **966** linhas de código.

7.5. Base de Dados

A base de dados foi construída através dos modelos do Django²⁷, que são classes Python que definem a estrutura de uma tabela na base de dados. Cada atributo existente numa classe representa um campo da tabela correspondente. O Django permite definir para cada atributo o

²⁷ <https://docs.djangoproject.com/en/5.0/topics/db/models/>

tipo de dados pretendido, assim como a definição de chaves estrangeiras para serem utilizadas nas relações entre tabelas (ver Figura 33).

Após criadas as classes e os respectivos atributos, é necessário executar dois comandos, de maneira a criar e aplicar as migrações necessárias. O primeiro comando é “**python manage.py makemigrations**”, que cria os arquivos de migração, necessários para alterar a estrutura da base de dados. O segundo comando é “**python manage.py migrate**”, que aplica as migrações criadas com o comando anterior, atualizando a estrutura da base de dados de maneira a ficar de acordo com os modelos definidos no projeto Django. Estes passos têm de ser repetidos sempre que são feitas alterações nesses modelos, para ver essas alterações refletidas na base de dados.

Os modelos Django permitem definir vários tipos de relações, como um-para-um, um-para-muitos, muitos-para-muitos e chave estrangeira. Isto permite definir as relações entre as diferentes tabelas da base de dados, necessárias em qualquer base de dados relacional.

Na Figura 34, é apresentada a estrutura da base de dados relacional PostgreSQL | PostGIS do sistema gWater.

```
248 class LayersCmems(models.Model):
249     """
250     CMEMS Layer
251     """
252
253     gpkg = models.CharField(max_length=50)
254     layer = models.CharField(max_length=500)
255
256     datehour = models.DateTimeField()
257
258     ia = models.ForeignKey(
259         InterestArea, on_delete=models.CASCADE,
260         db_index=True, db_column='ia',
261         related_name='cmemsia'
262     )
263     var = models.ForeignKey(
264         Variables, on_delete=models.CASCADE,
265         db_index=True, db_column="var"
266     )
267
268     added = models.DateTimeField()
269
270     step = models.IntegerField(default=0)
271
272     design = models.CharField(max_length=100)
273     work = models.CharField(max_length=20)
274     store = models.CharField(max_length=50)
275     glyr = models.CharField(max_length=50)
276
277     prcdid = models.CharField(max_length=20)
278     rstname = models.CharField(max_length=50)
```

Figura 33 - Modelo Django utilizado para a criação da tabela que guarda os dados associados às camadas CMEMS.

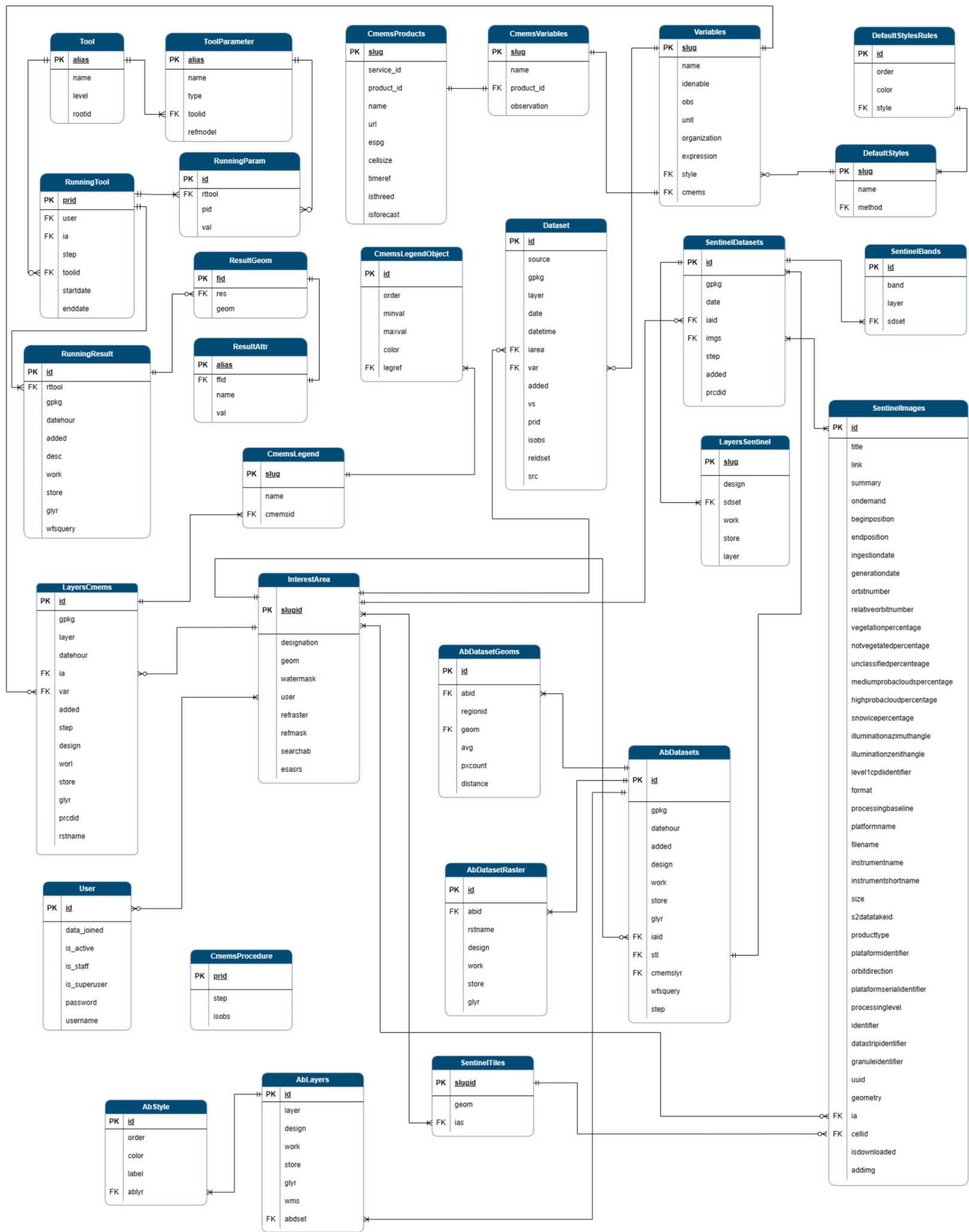


Figura 34 - Estrutura da Base de Dados Relacional do sistema gWater.

A tabela “**InterestArea**” atua como peça central nesta estrutura de base de dados, uma vez que todos os tipos de dados existentes no sistema estão associados a uma área de interesse. Esta tabela contém todas as extensões geográficas nas quais os utilizadores podem visualizar dados, estando também associadas à conta de cada utilizador.

Um exemplo de dados associados a esta tabela são as camadas do CMEMS, através da tabela “**LayersCmems**”. De notar que uma área de interesse pode ter zero ou muitas camadas CMEMS associadas a si, porém uma camada CMEMS só poderá corresponder a uma e apenas uma área de interesse. Estas relações são detalhadas em todo o esquema apresentado na Figura 34.

A tabela “**LayersCmems**” encontra-se ainda relacionada com as tabelas “**Variables**” e “**CmemsLegend**”. A relação com a tabela “**Variables**” permite associar cada camada a uma variável específica do sistema. É importante distinguir a tabela “**Variables**” da tabela “**CmemsVariables**”, sendo que a primeira diz respeito a todas as variáveis que possam existir no sistema, independentemente dos produtos e variáveis relacionadas com os dados produzidos pelo CMEMS. Cada camada CMEMS está associada a uma e apenas uma variável, porém cada variável poderá ter zero ou muitas camadas associadas. As tabelas “**CmemsLegend**” e “**CmemsLegendObject**”, permitem guardar os valores associados às legendas, contendo os valores dos intervalos e o código hexadecimal da cor associada a cada intervalo. Cada camada CMEMS poderá ter uma ou mais legendas associadas, uma vez que é possível a alteração da simbologia, sendo criada uma legenda na tabela “**CmemsLegends**”. Por sua vez, cada legenda CMEMS tem um ou mais registos da tabela “**CmemsLegendObject**” associados, sendo que cada registo desta tabela corresponde a um dos intervalos definidos. Cada registo da tabela “**CmemsLegendObject**” está associado a apenas um registo da tabela “**CmemsLegends**”, e cada registo da tabela “**CmemsLegends**” está associado a apenas um registo da tabela “**CmemsLayers**”, uma vez que cada legenda faz sentido apenas para uma determinada camada.

Outra tabela com relações com as áreas de interesse é a “**SentinelTiles**”, que serve para relacionar cada “bloco” de imagens Sentinel-2 na sua extensão original com as áreas de interesse existentes no sistema gWater, uma vez que uma área de interesse pode precisar de utilizar diferentes blocos de imagens Sentinel-2. Por outro lado, cada imagem do Sentinel-2 pode ser utilizada por mais que uma área de interesse. Além da “**SentinelTiles**”, a tabela “**SentinelImages**” também está associada com as áreas de interesse. Esta tabela serve para guardar os dados associados às imagens originais do Sentinel-2, estando também relacionada com as tabelas “**SentinelTiles**” e “**SentinelDatasets**”. A tabela “**SentinelDatasets**” é utilizada para guardar os dados associados às camadas Sentinel-2 que são utilizadas no sistema, após um pré-processamento das imagens originais para estarem de acordo com as áreas de interesse. Esta tabela está associada à tabela

“**SentinelBands**”, que guarda os dados associados às bandas de cada “**SentinelDataset**”. De notar que cada registo da tabela “**SentinelDatasets**” pode estar associado a um ou mais registos da tabela “**SentinelBands**”. A tabela “**LayersSentinel**”, serve para guardar os dados de cada camada associada a um *dataset* específico de camadas Sentinel-2.

A tabela “**CmemsProcedure**” é utilizada apenas nos processos de recolha e tratamento dos dados do CMEMS, servindo para identificar os conjuntos de dados, não estando associada a qualquer outra tabela.

A tabela “**AbDatasets**” e as restantes tabelas associadas a si, servem para armazenar todos os dados relacionados com as camadas resultantes dos *scripts* que fazem a identificação de elevados níveis de clorofila (algas) de forma automática.

Quanto às tabelas associadas à execução de ferramentas, e uma vez que já foi feita uma explicação das relações e finalidade de grande parte delas na secção anterior, falta apenas mencionar as tabelas “**ResultGeom**” e “**ResultAttr**”, que servem para armazenar os dados associados às camadas WFS resultantes dessas execuções. A primeira armazena as geometrias e a segunda armazena os atributos (dados alfanuméricos) associados.

7.6. Distribuição do Trabalho

Uma vez que o processo de implementação foi um esforço conjunto de uma equipa com três elementos, sendo uma delas o mestrando, é aqui apresentada a distribuição do trabalho realizado, de forma a clarificar quais as contribuições do mestrando ao longo da construção do sistema gWater.

Quanto à API, apenas os *endpoints* relacionados com a gestão e autenticação de utilizadores, gestão de camadas, *stores*, *workspaces* e estilos do GeoServer e pedidos de WMS e WFS ao GeoServer foram reutilizados de outros projetos desenvolvidos pelo outro membro da equipa de desenvolvimento. Além desses *endpoints* reutilizados, o outro membro da equipa teve participação única no desenvolvimento dos *endpoints* que fazem a gestão das variáveis e das áreas de interesse. Os restantes *endpoints* foram desenvolvidos pelo mestrando. Alguns exemplos de grupos²⁸ de *endpoints* criados pelo mestrando são a gestão das camadas CMEMS, dos estilos das camadas CMEMS, das legendas das camadas CMEMS, dos datasets e camadas do Sentinel-2, dos datasets e

²⁸ A expressão “grupos de *endpoints*” faz referência aos *endpoints* relacionados com recursos de uma determinada tabela e em que se utilizam os diferentes métodos HTTP para realizar as operações CRUD (Criar, Ler, Atualizar e Eliminar).

camadas *Algae Bloom*, das ferramentas, dos parâmetros das ferramentas, e dos resultados das ferramentas. Além dos *endpoints* que fazem a gestão destes recursos, foram também desenvolvidos pelo mestrando os *endpoints* que permitem a criação dos gráficos, assim como a criação das definições Python que permitem a extração dos dados das imagens.

As tabelas de Base de Dados criadas através dos modelos do Django, tiveram também participação conjunta no seu desenvolvimento, tendo sofrido bastantes alterações por parte dos desenvolvedores ao longo do processo de desenvolvimento.

Quanto aos *scripts* de recolha dos dados CMEMS, o mestrando apenas não contribuiu nos primeiros dois passos, que são o *download* e o pré-processamento dos dados, tendo contribuído de forma significativa nas etapas seguintes (ver Figura 20 e Figura 21). São elas: a criação dos estilos na base de dados, o envio para o servidor, a reclassificação das imagens e envio do resultado para o servidor, a criação dos serviços de mapas, a criação de ficheiros de estilo e associação aos estilos no GeoServer e a eliminação dos dados desnecessários. Todos estes *scripts* ficaram ao encargo do mestrando, tendo sido efetuadas algumas correções pela restante equipa posteriormente.

Relativamente aos *scripts* de recolha dos dados do Sentinel-2, as contribuições do mestrando foram feitas após o terceiro passo da segunda fase, que cria o mosaico com as bandas em formato tif (ver Figura 22), tendo sido o principal responsável pela sua implementação. Uma vez que estes *scripts* também utilizam definições Python para manipular os dados geográficos, o mestrando participou igualmente no seu desenvolvimento. Um exemplo desse desenvolvimento por parte do mestrando é a definição Python que cria um ficheiro *raster* composto com as bandas necessárias para criar posteriormente diferentes combinações que permitem visualizar as imagens Sentinel-2 em cor natural e em falsa cor na interface gráfica do sistema gWater, tal como explicado na secção 7.1.2.

Os *scripts* que permitem a identificação de algas através das imagens do Sentinel-2 de forma automática, não são apresentados neste capítulo, uma vez que o mestrando não teve nenhuma intervenção significativa na sua implementação. Contudo, o mestrando é responsável pela apresentação dos resultados criados por esses *scripts* na aplicação de cliente.

Na aplicação de cliente, a construção da interface gráfica ficou sobretudo ao encargo do mestrando, desenvolvendo as funcionalidades que permitem a visualização das camadas, alteração de simbologia, catálogo dos dados, adição dos dados ao “Map Layers”, construção de gráficos e execução de ferramentas. Outro membro da equipa contribuiu também na implementação da arquitetura Redux através da biblioteca NgRx, permitindo reduzir a complexidade de circulação dos dados entre as diferentes componentes, uma vez que a fonte de dados passa a ser a mesma, que é a chamada *Store*, tal como explicado na secção 7.3.3.

Nos *scripts* que permitem a execução de ferramentas, houve também uma participação conjunta no seu desenvolvimento.

8. Resultados e Discussão

Os resultados obtidos são apresentados neste capítulo através de figuras que demonstrem o aspeto e funcionamento do sistema, como a visualização de imagens de diferentes fontes no mapa, criação de gráficos e execução de ferramentas.

A apresentação do resultado da implementação do sistema começa com a página de *login* (Figura 35), onde os utilizadores fazem a sua autenticação perante o sistema. Nesta página existe um formulário com dois campos. O primeiro serve para introduzir o *email* e o segundo para introduzir a palavra-passe. Por baixo do botão de *login*, existe um segundo botão, que serve para recuperar a palavra-passe em caso de esquecimento da mesma.

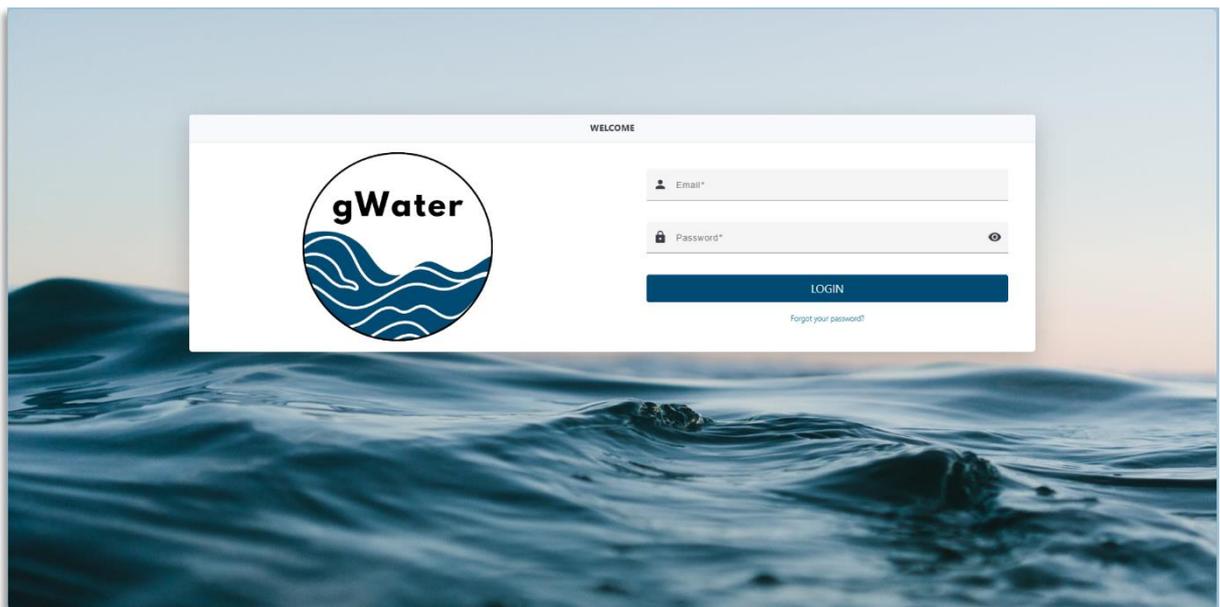


Figura 35 - Página de login do sistema gWater.

Na Figura 36, é apresentada a página principal do sistema gWater. É para esta página que o utilizador é redirecionado após fazer a autenticação na página de *login*. A barra de navegação apresenta três opções e um *icon* junto ao lado direito que serve para apresentar mais opções, sendo uma delas o *logout*. A opção selecionada é a opção “Home”, que é a página principal. As outras duas

opções, “Control Panel” e “Dashboard”, encontram-se visíveis nesta barra de navegação apenas para efeito estético e de demonstração, uma vez que não foram implementadas. A opção “Control Panel” funcionaria como página de gestão dos utilizadores, tanto do superutilizador como do utilizador comum, sendo que no caso do superutilizador esta página apresentará um maior número de opções e funcionalidades.

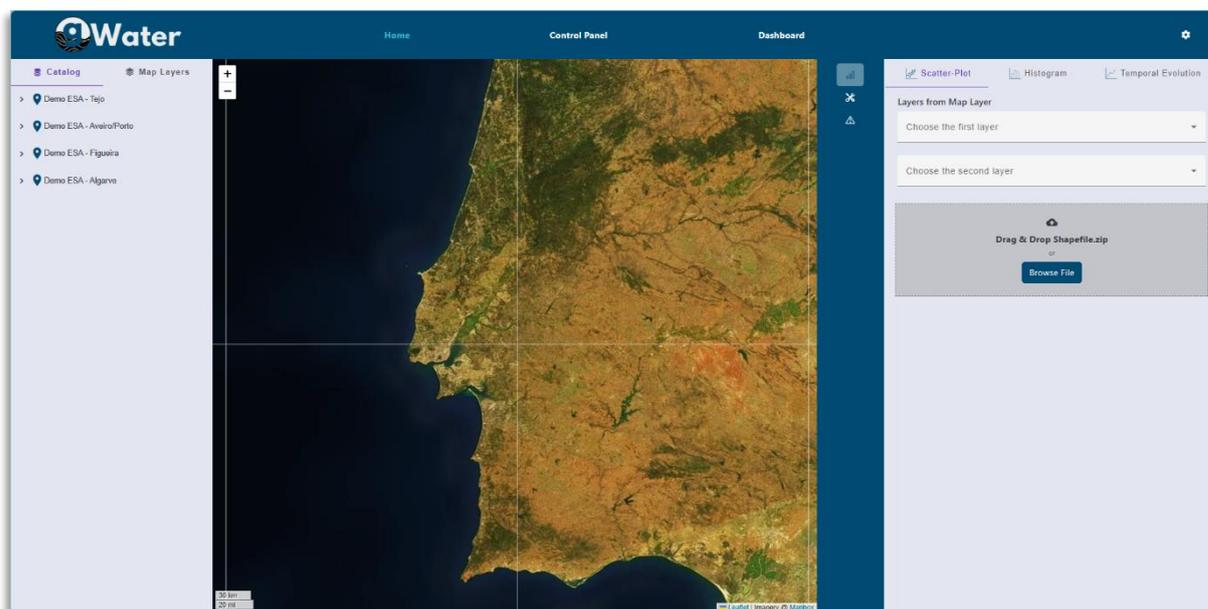


Figura 36 - Página principal do sistema gWater.

Do lado esquerdo da página, existe um menu lateral com duas opções, “Catalog” e “Map Layers”. O utilizador pode alternar entre as duas opções de maneira a pesquisar os dados no “Catalog” e adicioná-los ao “Map Layers”. Na Figura 37, é demonstrado o aspeto do catálogo de dados após a expansão de alguns dos nós da estrutura em árvore. É visível a hierarquia da árvore descrita anteriormente neste relatório. O primeiro nó corresponde à área de interesse, o segundo corresponde à fonte dos dados e o terceiro corresponde à variável. O terceiro nó apenas existe quando a fonte selecionada for “CMEMS Variables”, sendo que nos restantes casos aparecem logo as camadas disponíveis.

À frente de cada camada disponível aparece um *icon* que serve para adicionar a camada ao “Map Layers”.

Na Figura 38, é visível a existência de duas camadas no “Map Layers” de duas áreas de interesse distintas. As duas camadas correspondem a dados do CMEMS e representam duas das variáveis disponíveis, a clorofila e a temperatura. É possível visualizar a legenda de cada uma das camadas, uma vez que se encontram expandidas. As cores de cada uma correspondem à cor definida para cada variável, sendo atribuída a cada camada no momento da sua criação, que acontece nos *scripts* de recolha e processamento dos dados. Como já foi dito, é possível alterar a simbologia, atribuindo a rampa de cores desejada.

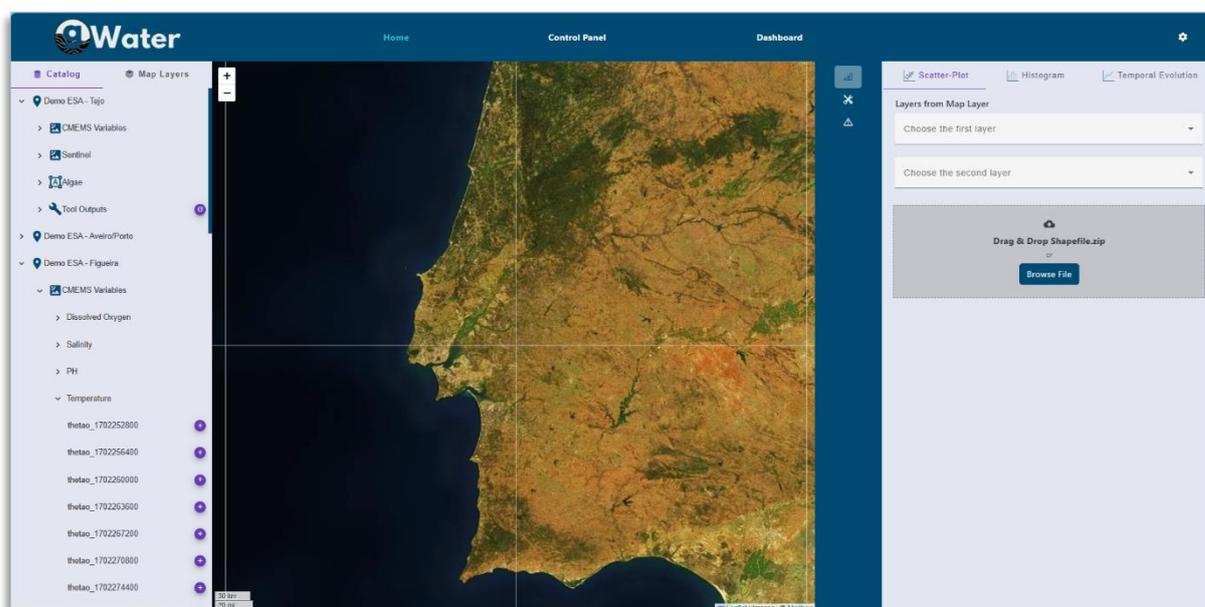


Figura 37 - Página principal com vários nós do catálogo de dados abertos.

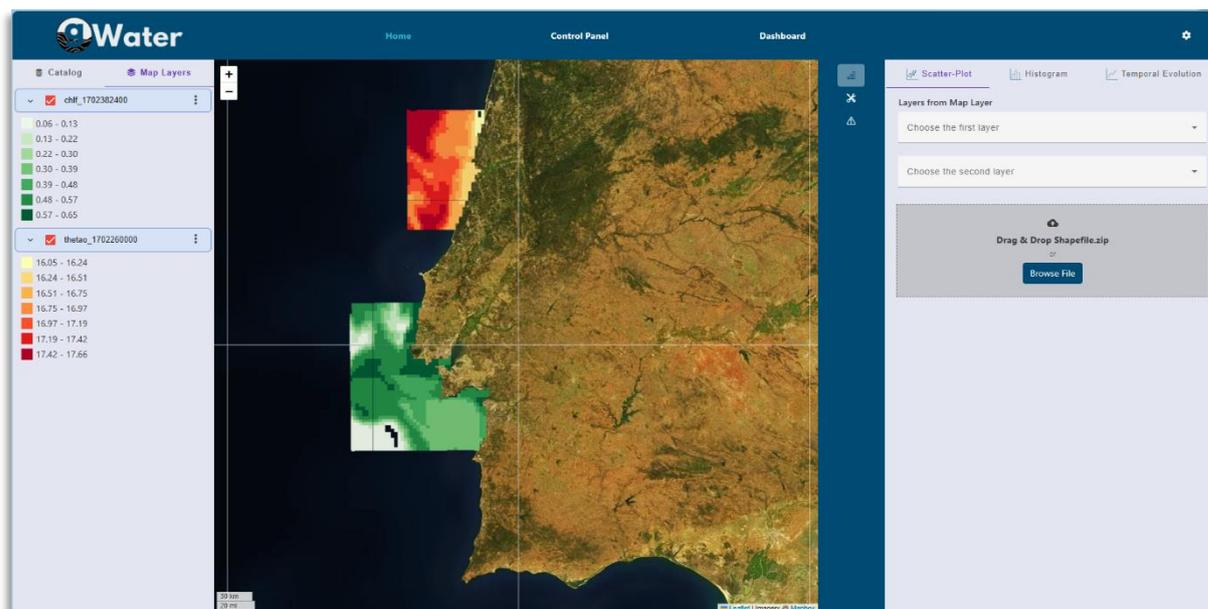


Figura 38 - Página principal com duas camadas ativas.

De seguida, são apresentados na Figura 39 e Figura 42 respetivamente, um histograma e um gráfico de dispersão, criados a partir das camadas existentes no “Map Layers”. No caso do histograma, o utilizador escolhe a camada e o número de intervalos pretendido, oferecendo ao utilizador a possibilidade de personalizar o gráfico conforme o seu desejo (ver Figura 40). No caso da camada escolhida ser do tipo Sentinel-2, o utilizador pode escolher um intervalo de corte, permitindo ocultar a cauda do histograma que é comum nesse tipo de dados e que pode prejudicar a interpretação do gráfico (ver Figura 41). Além disso, no caso de camadas Sentinel-2, o utilizador pode ainda indicar a amplitude de cada intervalo ao invés de indicar o número de intervalos, e ainda a banda na qual deseja ler os valores, permitindo assim fazer várias análises deste tipo de dados.

Quanto ao gráfico de dispersão, este pode ser obtido através da seleção de duas camadas do tipo CMEMS e do *upload* de um ficheiro com pontos na área de interesse desejada. O ficheiro tem de ser um *shapefile* comprimido em formato zip. Além disso, as camadas de interesse escolhidas devem pertencer à mesma área de interesse, caso contrário o sistema apresenta uma mensagem ao utilizador a indicar que os dados introduzidos não estão corretos, ainda antes de fazer qualquer chamada à API. O gráfico mostrará os valores das duas variáveis escolhidas nos pontos do ficheiro que foi importado. Em ambos os gráficos são apresentados os valores ao longo dos eixos correspondentes aos valores existentes nos dados, assim como o tipo de variável (e respetiva unidade) a que se refere. Para obter informação com maior precisão sobre os valores dos gráficos, o utilizador pode passar o ponteiro do rato sobre o gráfico e visualizar uma etiqueta com os valores

nesse local. No caso do histograma, é visualizado o intervalo de valores da barra para que está a apontar e o número da frequência desses valores. No caso do gráfico de dispersão, são apresentadas as coordenadas do ponto de acordo com os valores presentes nos respectivos eixos.

Ainda relativamente ao gráfico de dispersão, o *upload* do ficheiro pode ser feito através de arrasto de um ficheiro ou do clique no botão “Browse File”, que abre uma janela do explorador do sistema operativo que permite procurar o ficheiro desejado. Caso o ficheiro seja aceite, o fundo da zona de arrasto do ficheiro fica a verde, como é visível na Figura 42, caso contrário ficará a vermelho, indicando o erro sucedido.

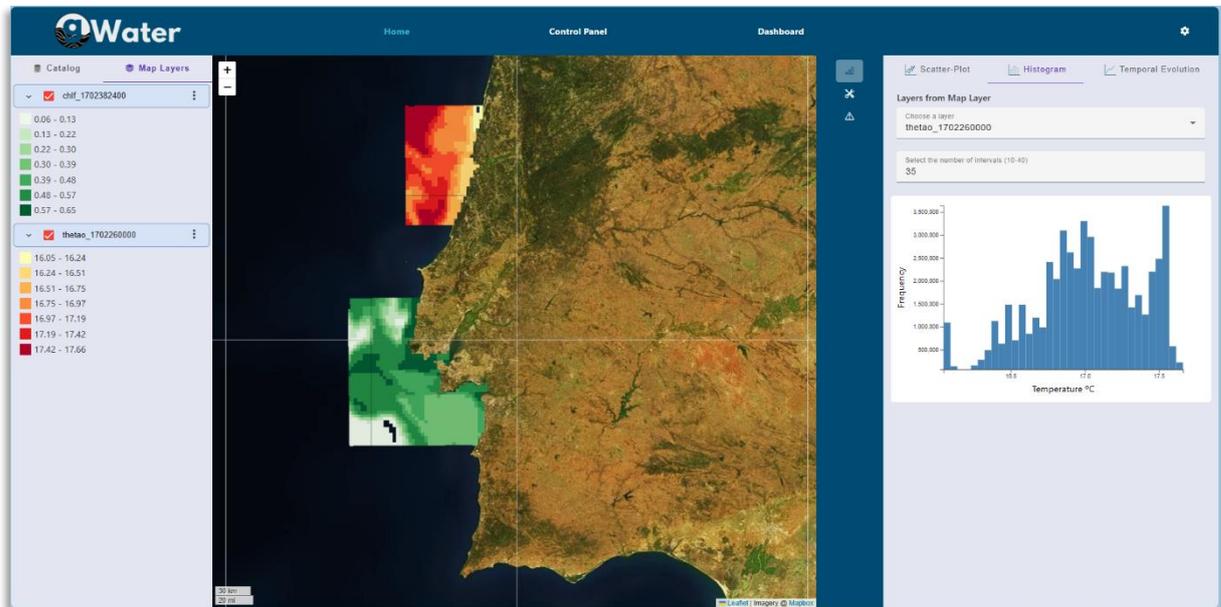


Figura 39 - Página principal com um histograma.

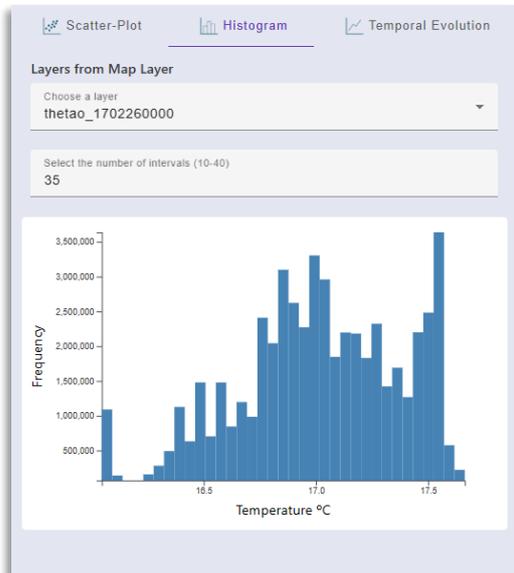


Figura 40 - Histograma criado no sistema gWater através de dados CMEMS.



Figura 41 - Histograma criado no sistema gWater através de dados Sentinel-2.

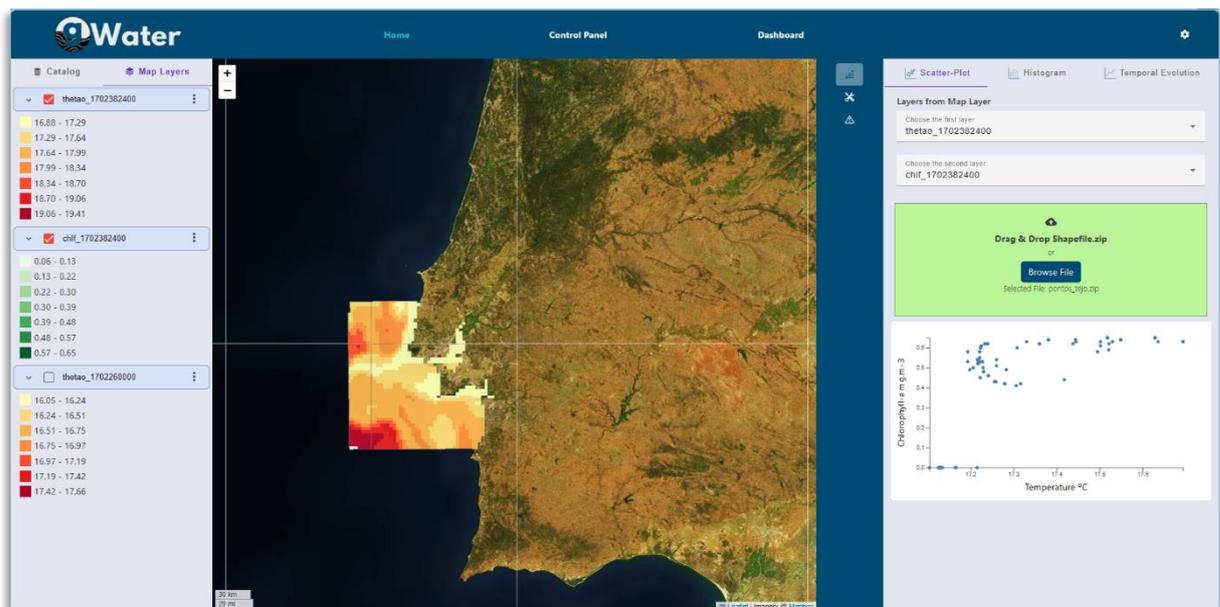


Figura 42 - Página principal com um gráfico de dispersão.

Existe ainda o gráfico de evolução temporal, que permite visualizar a evolução das diferentes variáveis do CMEMS num ponto específico dentro de uma área de interesse.

Apesar de não ser aqui apresentado na sua forma final por não haver dados de diferentes datas, é apresentado na Figura 43 o menu de seleção dos dados para a sua construção. O facto de não haver dados para diferentes datas, deve-se a limitações de *hardware* do equipamento utilizado, uma vez que as execuções dos *scripts* de recolha de dados requerem um uso considerável de memória e os dados produzidos consomem muito espaço de armazenamento. Dessa forma, estes *scripts* foram executados apenas uma vez, para obter dados que permitissem testar o funcionamento dos mesmos e das restantes funcionalidades do sistema.

Para poder visualizar o gráfico, o utilizador tem de seleccionar a área de interesse, escolher uma variável dos dados do CMEMS, um intervalo temporal e importar um ficheiro com um ponto, de maneira a obter os valores das imagens de diferentes datas nesse ponto. Poderia ser omitida a escolha da área de interesse, uma vez que pode ser obtida através da localização do ponto. No entanto, essa abordagem implicaria mais processamento do lado do servidor, o que levaria a tempos de espera mais longos. Desta forma, a lógica existente na API apenas verifica se é possível obter o valor das camadas da área seleccionada nesse ponto quando tenta obter esse valor. Além disso, ao indicar a área de interesse e a variável, o sistema sabe logo quais os dados que tem de utilizar, sem necessidade de fazer mais verificações.

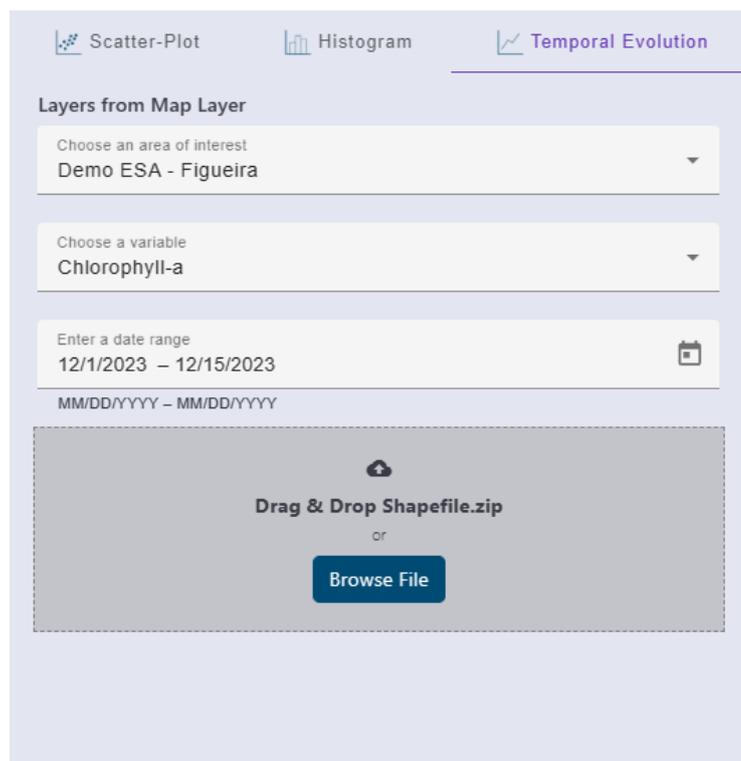


Figura 43 - Menu que permite obter o gráfico de evolução temporal.

Os tempos de espera para a obtenção dos gráficos variam conforme o tamanho dos dados a usar. Por exemplo, segundo as áreas de interesse das camadas visíveis no mapa na Figura 38, que foram criadas durante a fase de implementação, caso a camada escolhida seja da área de interesse correspondente à zona da foz do rio Mondego, o tempo de espera é menor. No caso da zona da foz do Tejo, o tempo de espera é maior, uma vez que a área de interesse é maior. No entanto, foram adotadas medidas que permitem dar mais ou menos ênfase à performance caso o *hardware* do equipamento onde o sistema ficará hospedado assim o permita, tal como foi explicado no capítulo anterior. Estas medidas estão associadas à leitura dos dados em blocos, reduzindo assim a utilização de memória caso seja necessário, mas aumentando o tempo de espera. Contudo, é possível arranjar um equilíbrio que permita obter um bom desempenho sem consumir muitos recursos. De notar que nada disto está relacionado com o equipamento dos utilizadores do sistema, mas sim *hardware* disponível no servidor, no entanto, isto pode ser um aspeto importante para a experiência dos utilizadores aquando da utilização destes recursos.

Na Figura 44, é apresentada a página principal com o catálogo de ferramentas aberto. Como já foi referido anteriormente, apenas uma ferramenta funciona, sendo que as restantes aqui apresentadas são apenas ilustrativas. Este catálogo de ferramentas funciona como uma estrutura em árvore, à semelhança de *softwares* SIG como o ArcGIS Pro, em que existe um agrupamento das

ferramentas consoante o seu tipo ou finalidade. Além de o utilizador poder navegar na estrutura em árvore, pode também pesquisar a ferramenta através do campo de procura disponível no topo.

Após efetuada a escolha da ferramenta (ver Figura 45), são apresentados os campos relativos aos parâmetros da mesma. Após o preenchimento destes campos, o utilizador deverá clicar no botão “Run” no canto inferior direito, que dá início ao processamento da ferramenta. Ao longo do processamento, são apresentadas mensagens relativas ao progresso da execução da ferramenta, tal como já foi referido anteriormente (ver Figura 46).

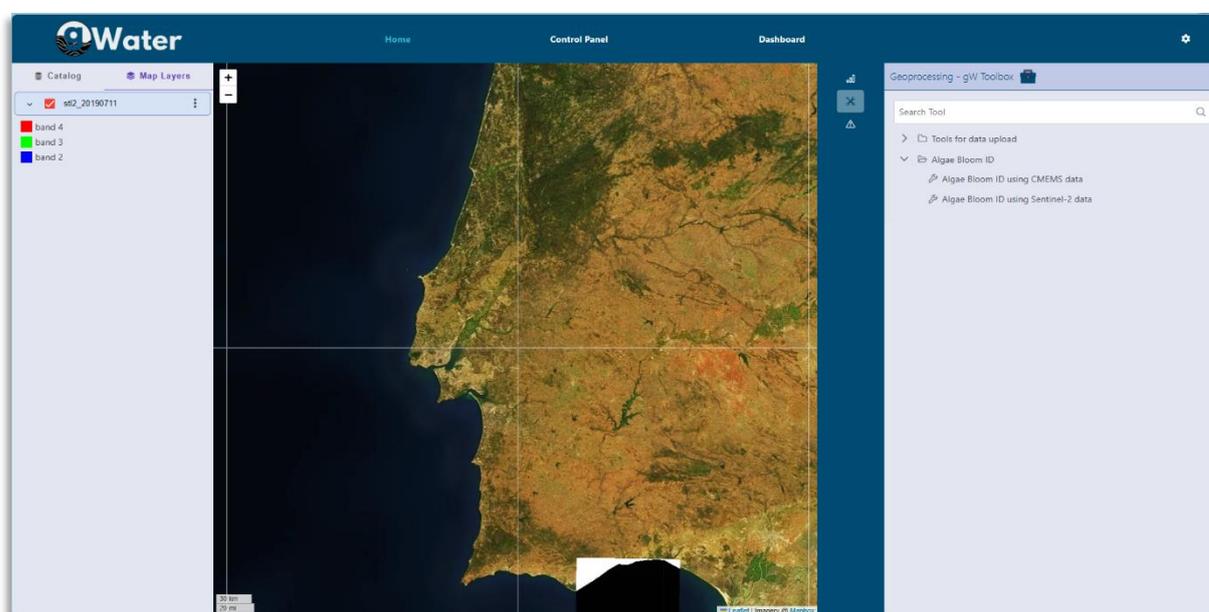


Figura 44 - Página principal com o catálogo das ferramentas aberto do lado direito da página.

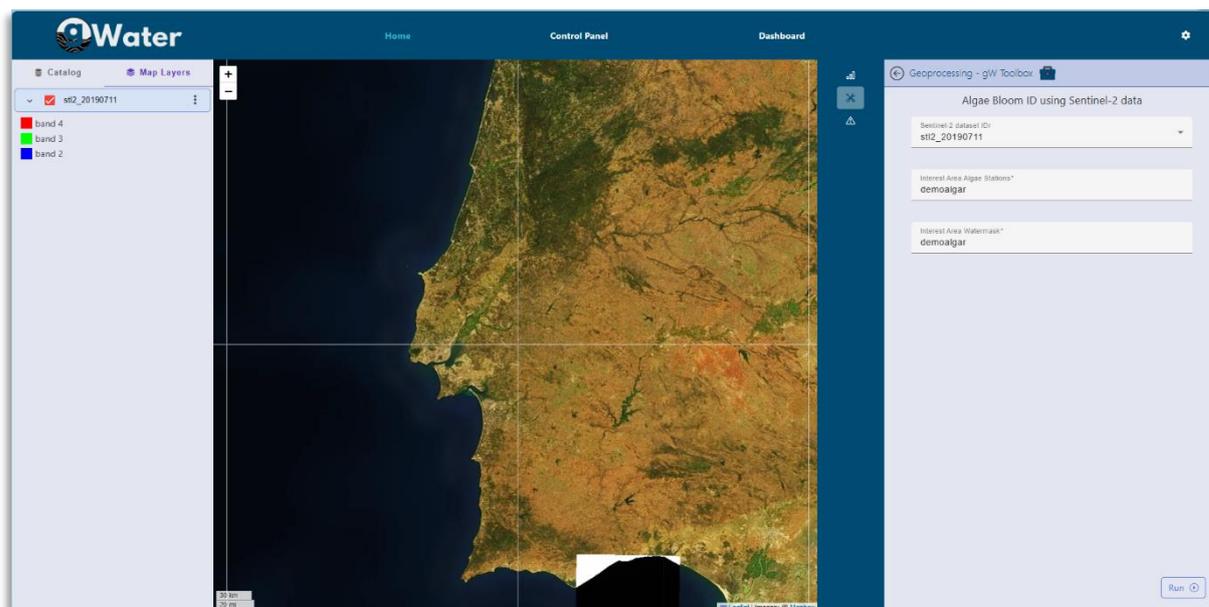


Figura 45 - Página principal com uma ferramenta aberta.

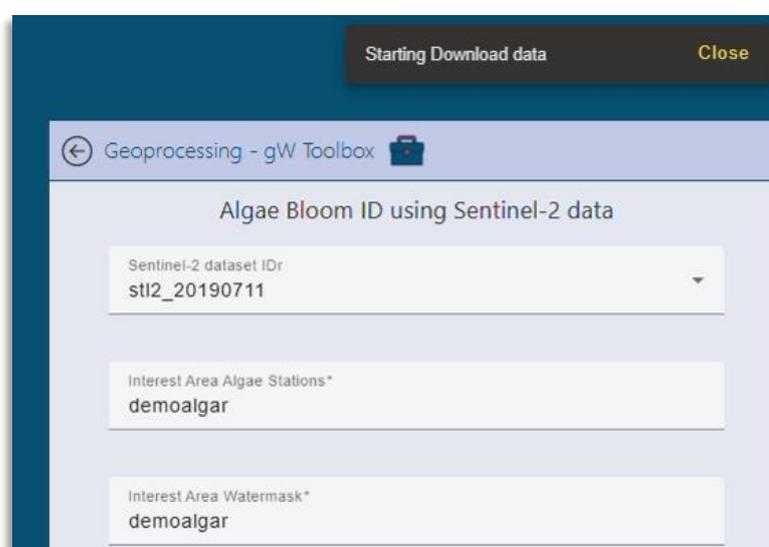


Figura 46 - Exemplo de indicação do progresso da execução de uma ferramenta.

Além das fontes de dados CMEMS e Sentinel-2, existem também dados relacionados a concentrações de clorofila, designados no catálogo por “Algae” (ver Figura 37), e representados através de polígonos, como demonstra a Figura 47. Estes polígonos são gerados automaticamente através de *scripts* que fazem o processamento das imagens Sentinel-2. Estes *scripts* foram criados

como exemplo de ferramenta que é executada automaticamente para criar áreas de ocorrências de valores anormais para qualquer variável, ficando também disponíveis no catálogo. Neste caso, foi utilizada a variável clorofila, que poderá indicar a presença de algas. De notar que o mestrando apenas colaborou na representação dos dados produzidos por estes *scripts*, não estando por isso detalhados no capítulo da implementação.

No entanto, o utilizador tem a possibilidade de executar a ferramenta que produz exatamente os mesmos resultados, uma vez que tanto os *scripts* discutidos no parágrafo anterior como a ferramenta disponível atualmente no sistema se baseiam nas mesmas definições Python que realizam esse processamento. Desta forma, o resultado da execução da única ferramenta disponível no sistema atualmente (ver Figura 48) é semelhante ao resultado dos *scripts* discutidos no parágrafo anterior.

O ideal seria o utilizador poder fazer o *upload* de imagens para o sistema e executar esta ferramenta para qualquer imagem. Dessa forma, não estaria dependente dos dados existentes no sistema. Além disso, essa funcionalidade aumentaria a flexibilidade do sistema gWater.

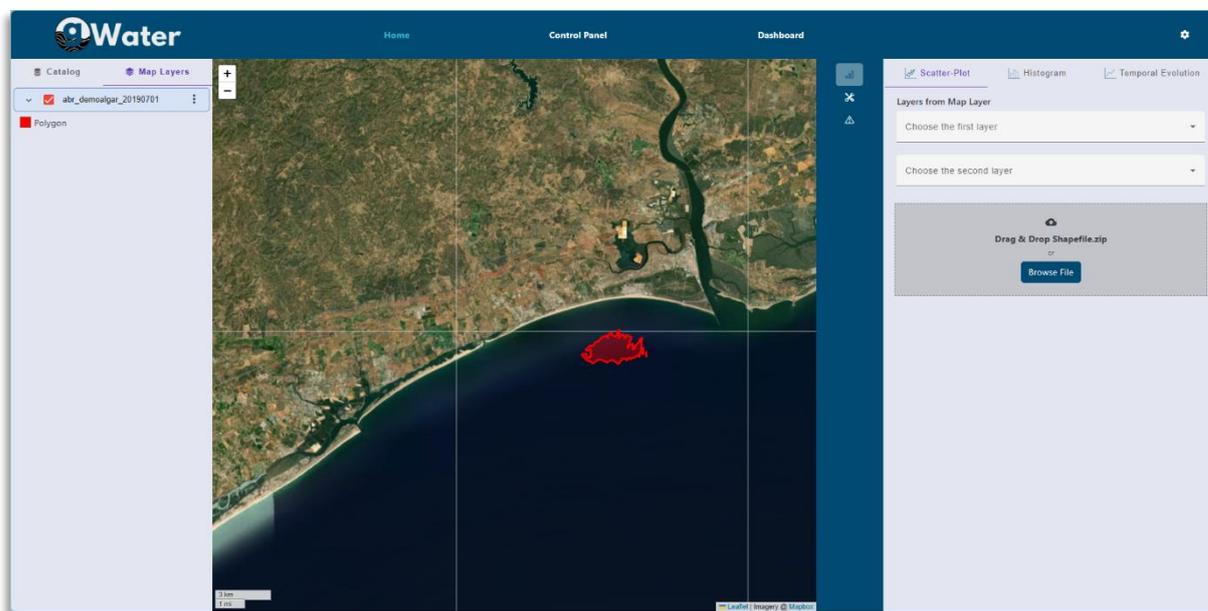


Figura 47 - Página principal com uma camada WFS relacionada com o aparecimento de algas, resultado da execução dos *scripts* que realizam este processamento de forma automática.

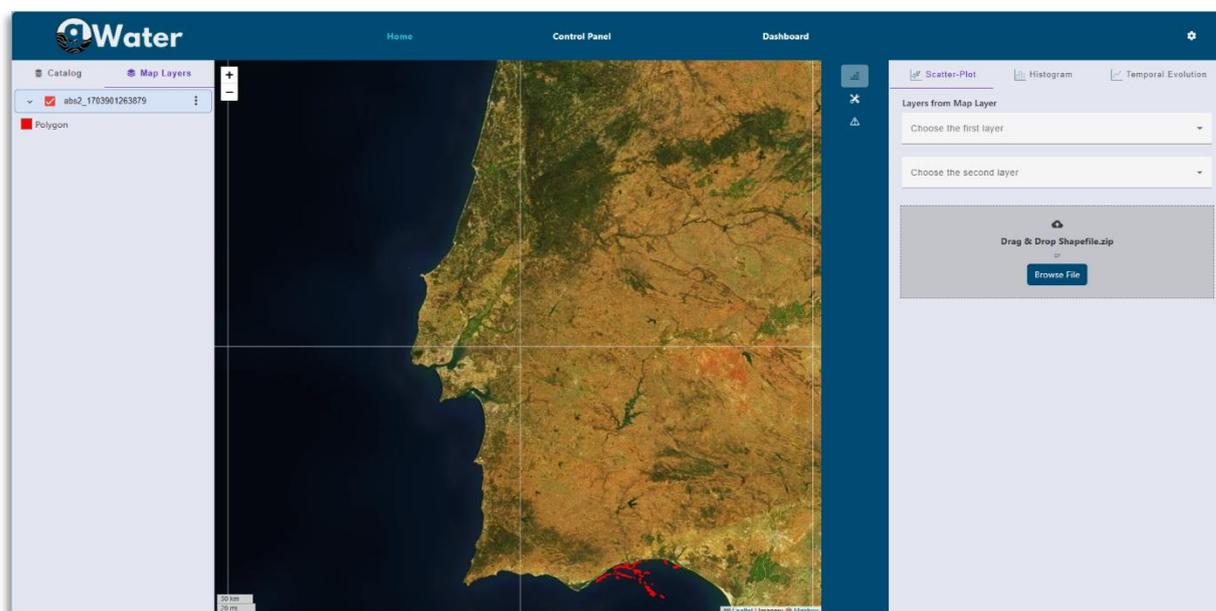


Figura 48 - Exemplo de camada WFS relacionada com a ocorrência de algas, resultado de uma execução de ferramenta.

Quanto à simbologia, é possível alterá-la através das opções existentes em cada camada do Map Layers. Para ver a lista de opções, basta clicar no botão com três pontos do lado direito de cada camada do Map Layers (ver Figura 49). Após escolher a opção “Change Symbology”, é aberta uma janela (ver Figura 50, Figura 51 e Figura 52), onde é possível escolher a simbologia desejada. Na Figura 50, vemos a alteração da simbologia de uma camada do tipo CMEMS, que oferece a possibilidade de escolher entre diversas rampas de cores. Apesar de haver um campo para a escolha do número de classes e outro para o método que deverá ser utilizado para fazer a divisão dos valores nesse número de classes, estes não se encontram em funcionamento, pelo que o resultado não tem atualmente em consideração os valores indicados nestes campos. No entanto, os métodos para o cálculo dos intervalos encontram-se implementados nas definições Python que fazem este processamento dos dados, sendo o motivo pelo qual não se procedeu à implementação completa desta funcionalidade a alteração que seria necessária fazer na estrutura da base de dados, que seria bastante complexa e consumiria bastante tempo. Além disso, foi planeada e discutida uma possível solução para resolver o problema existente causado pela forma como são guardados os estilos (cores) e valores dos intervalos de cada camada CMEMS. Essa solução passa pela eliminação da

tabela “DefaultStylesRules”, que guarda atualmente as cores de cada estilo existente no sistema. Cada cor de um determinado estilo existente na tabela “DefaultStyles”, é um registo na tabela “DefaultStylesRules”. Desta forma, estamos limitados ao número de registos existentes nesta tabela, uma vez que representam o número de classes que podemos utilizar na simbologia de uma camada. Ao eliminar esta tabela, podemos guardar apenas a cor inicial e final de uma escala de cores na tabela “DefaultStyles” e utilizar funções para obter as cores existentes entre essas duas cores, fornecendo também o número de cores que queremos obter, que correspondem aos intervalos desejados. Uma vez que é necessário haver um estilo e um número de classes padrão, deverá ser guardado na tabela “Variables” o número de intervalos desejados para cada variável. Esse número poderá ser também guardado na tabela “DefaultStyles” ou então definido numa variável nos *scripts* que fazem a recolha e tratamento dos dados, indicando o número de classes pretendido no momento da criação desses estilos padrão. De notar que atualmente a tabela “Variables” guarda o estilo associado a cada variável. É por isso que as camadas associadas à clorofila aparecem por padrão com uma escala de cores em tons de verde e as camadas associadas à temperatura aparecem com tons de laranja. Na aplicação de cliente já é apresentado um gradiente de cores quando se pretende alterar a simbologia de uma camada CMEMS (ver Figura 50), uma vez que foi criada uma função que cria esse gradiente de forma automática, através da cor inicial e cor final. Contudo, são utilizadas a primeira e a última cor do estilo padrão, que neste momento são sempre sete. Isto apenas serve para não mostrar sete quadrados com as cores correspondentes e dar a ideia do que se pretende fazer, que é obter cores ao longo desse gradiente para o número de classes desejado.

Quanto à alteração da simbologia de uma camada Sentinel-2, é apenas possível escolher entre duas combinações de bandas possíveis, criadas aquando da recolha dos dados, tal como foi referido no capítulo anterior. A interface para a alteração da simbologia das camadas Sentinel-2 é exibida na Figura 51.

Para as camadas WFS relacionadas com a ocorrência de elevados níveis de clorofila, é apenas possível alterar a cor dos polígonos (ver Figura 52).

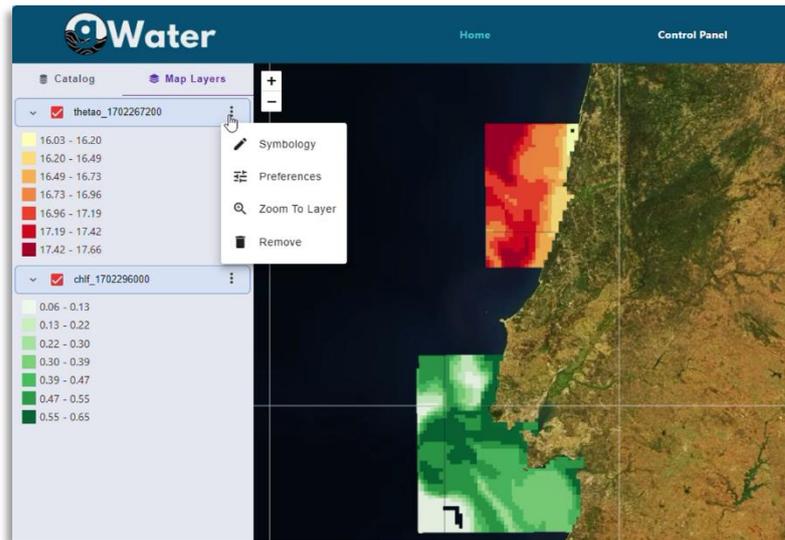


Figura 49 - Opções disponíveis em cada camada do Map Layers.

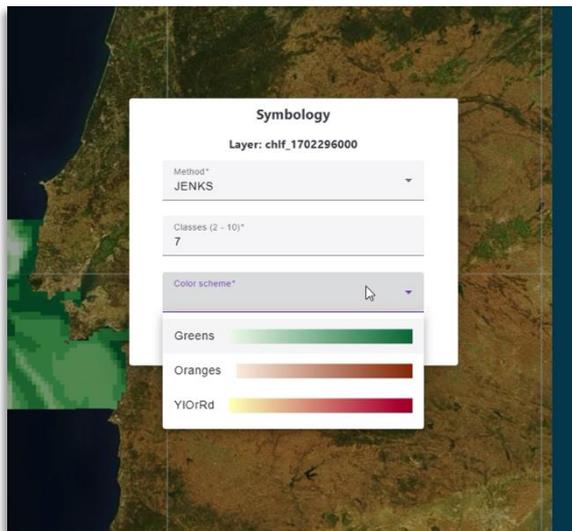


Figura 50 - Janela que permite alterar a simbologia de uma camada CMEMS.

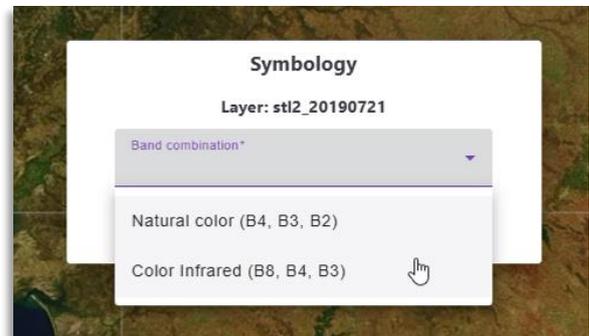


Figura 51 - Janela que permite alterar a simbologia de uma camada Sentinel-2.

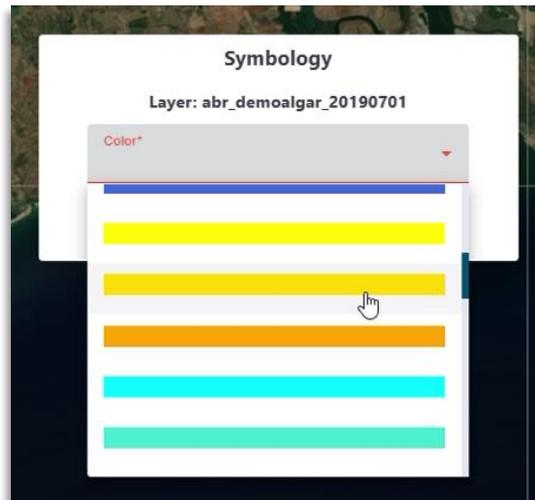


Figura 52 - Janela que permite alterar a simbologia de uma camada relacionada com a ocorrência de algas.

Além da simbologia, existem ainda as preferências de visualização de cada camada (ver opções na Figura 49), que permite alterar a opacidade da camada e, no caso das camadas CMEMS, permite também definir intervalos de corte de maneira a filtrar a imagem de acordo com as necessidades específicas de visualização do utilizador (ver Figura 53).

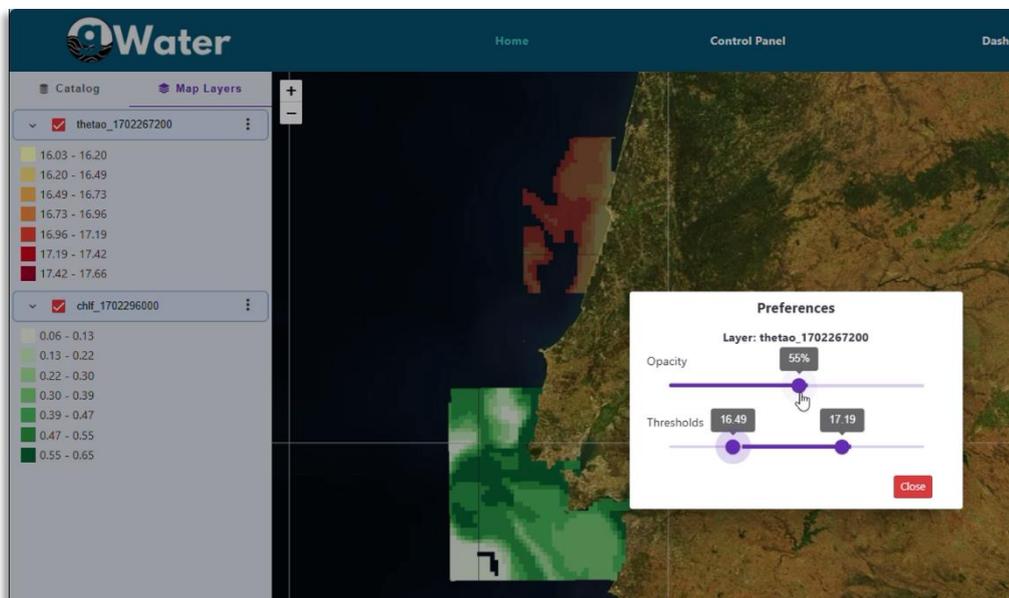


Figura 53 – Janela das preferências de visualização de uma camada.

Os resultados apresentados nesta secção demonstram que o objetivo principal do trabalho proposto foi alcançado, uma vez que o sistema aqui apresentado consegue recolher, tratar,

processar e apresentar informação geográfica relacionada com a monitorização da qualidade da água em ambiente marinho. Os utilizadores deste sistema conseguem obter informações acerca do estado da água nas diferentes áreas de interesse, através dos diversos parâmetros disponíveis nos dados do CMEMS. Além disso, conseguem visualizar áreas onde poderá haver ocorrências de algas, através de polígonos representativos das mesmas. Para obter mais informações associadas às frequências dos valores de cada imagem ou da relação entre diferentes imagens em determinados pontos, o utilizador pode ainda criar os gráficos necessários para tal, nomeadamente histogramas e gráficos de dispersão. Tal como já foi referido, o gráfico de evolução temporal, apesar de ter sido desenvolvido não é aqui apresentado.

O sistema gWater permite ainda a execução de ferramentas de geoprocessamento. Apesar de neste momento apenas estar disponível uma ferramenta que permite a identificação de zonas com características propícias à ocorrência de *Algae Blooms*, é possível introduzir mais. Contudo, será necessário modificar os *scripts* que executam a sequência de processos necessários desde o clique do utilizador para executar a ferramenta até ao aparecimento dos resultados no catálogo de dados e a consequente visualização dos mesmos. Neste momento, o terceiro *script* dessa sequência está preparado para executar apenas a definição Python que, através de diversas bibliotecas que permitem manipular dados geoespaciais, efetuam a identificação das zonas com potenciais *Algae Blooms*. No entanto, uma possível solução para permitir a execução de diversas ferramentas seria implementar um *script* específico para cada ferramenta, enquanto o terceiro *script* executa o respetivo *script* associado à ferramenta desejada. Esta solução seria adequada para implementar outra funcionalidade não existente neste momento, que é a possibilidade de um utilizador importar o seu próprio *script*, à semelhança do que acontece nos diversos *softwares* SIG. Apesar de conhecer a solução, esta não foi implementada por falta de tempo para o seu desenvolvimento, contudo será uma possível melhoria a implementar futuramente no sistema.

Quanto às possíveis ferramentas que possam ser introduzidas no sistema, estas podem estar associadas a qualquer outro contexto, uma vez que é possível utilizar dados do Sentinel-2 para obter informações relevantes para qualquer outra área de aplicação. No entanto, e uma vez que o sistema desenvolvido tem como área de aplicação a monitorização da qualidade da água, o ideal será a introdução de mais ferramentas que permitam obter informações relevantes neste contexto. Isto significa que o sistema poderá ser totalmente transformado para ser utilizado noutra contexto completamente diferente, uma vez que a lógica de funcionamento será a mesma.

Relativamente à página de gestão dos utilizadores, não foi feita a implementação da interface gráfica na aplicação de cliente do sistema. Contudo, estão criados todos os *endpoints* que permitem realizar essa gestão, estando assim a API preparada para uma possível implementação da

interface gráfica no futuro. Além disso, foram criados diversos *Jupyter Notebooks* que permitem realizar essa gestão, como, por exemplo, editar uma camada do CMEMS, criar uma ferramenta ou eliminar uma camada do Sentinel-2. Estes *Jupyter Notebooks*, além de poderem servir de ferramenta alternativa de gestão dos dados do sistema no futuro, foram utilizados durante o processo de implementação para testar as diversas funcionalidades antes de implementar a interface gráfica.

Por último, ficou por realizar uma etapa importante na implementação deste WebSIG, que é a fase de testes, devido à falta de tempo para a sua realização. Esta fase é responsável por garantir que o sistema atende aos requisitos e atributos de qualidade definidos inicialmente e funciona conforme esperado. Estes testes podem ser divididos em testes de funcionalidade, usabilidade, desempenho, segurança, compatibilidade e integração, permitindo garantir a qualidade e conformidade com os requisitos do sistema, assim como a satisfação dos utilizadores. Desta forma, este sistema não está preparado para ser colocado em produção²⁹, uma vez que aspetos tão fundamentais como a segurança não foram devidamente testados. Além disso, não foram efetuados testes de desempenho, algo especialmente importante no contexto de um WebSIG, uma vez que requer o processamento e visualização de grandes conjuntos de dados. Neste seguimento, é também necessário haver um computador de grande capacidade de processamento para realizar todas as operações necessárias em simultâneo, tal como a recolha e tratamento de dados de forma automática, além de lidar com os pedidos dos utilizadores para a execução de ferramentas. Isto requer um investimento significativo a nível de *hardware*, e um espaço para o equipamento a funcionar de forma permanente.

Uma vez que é na fase de testes que se comprova que o sistema cumpre os atributos de qualidade definidos previamente, não é possível verificar se tal acontece.

Existem diversas estratégias para a realização desses testes. A caso de exemplo temos os testes de usabilidade, onde é necessário avaliar a experiência do utilizador quanto à utilização do sistema. Para isso, são selecionados vários participantes, idealmente parte do público-alvo do sistema em análise, e são atribuídas tarefas a cada um deles, sendo estas tarefas representativas das ações que os utilizadores reais possam realizar no futuro. Durante a realização deste teste, os moderadores observam e registam todo o processo, obtendo o *feedback* dos participantes.

No caso da segurança, os testes podem passar pela realização de pedidos à API sem a devida autorização ou permissão para aceder a determinados dados. Desta forma, veremos se a API se comporta da maneira esperada. Quanto à aplicação de cliente, podem ser feitos testes de

²⁹ Colocar o sistema em produção significa transferir o sistema do ambiente de desenvolvimento para o cenário real, onde se torna operacional e disponível para os utilizadores.

acesso a determinada página sem efetuar previamente o *login*. O comportamento esperado, será que a aplicação direcione esse utilizador para a página de *login*. Por outro lado, um utilizador que faça o *login* deverá ter acesso garantido ao sistema, consoante as suas permissões. Podem também ser feitos testes de invasão, através da utilização de ferramentas e estratégias normalmente utilizadas em ataques desta natureza.

Ainda sobre possíveis testes para verificar se os atributos de qualidade estão a ser cumpridos, podem ser feitas comparações entre os dados originais das diferentes fontes e os dados que o sistema apresenta, comprovando o cumprimento do atributo de qualidade associado à precisão do sistema.

9. Conclusões

Este estágio, teve como objetivo o desenvolvimento e implementação de um sistema capaz de fazer a monitorização da qualidade da água em ambiente marinho, com tecnologias de código aberto e utilizando dados do programa Copernicus, nomeadamente dados do CMEMS e Sentinel-2. Através dos requisitos funcionais, diagramas e tabelas de caso de uso e atributos de qualidade, foram definidas as funcionalidades e características necessárias. O destaque deste sistema é a página de visualização dos dados referidos acima, através de um mapa com diversas funcionalidades. Além disso, a possibilidade de execução de ferramentas é uma funcionalidade que diferencia este sistema dos restantes WebSIG existentes e disponíveis para visualização de dados georreferenciados relacionados com a qualidade da água.

O trabalho desenvolvido é o resultado de um esforço em equipa, tendo o mestrando sido um dos desenvolvedores.

O resultado, apesar de não satisfazer todos os requisitos propostos inicialmente, cumpre as funcionalidades base de qualquer WebSIG, através da visualização de dados num mapa que permite a interação do utilizador com o mesmo, manipulação de informação geográfica (ferramentas) através da interface gráfica e pela combinação de diferentes fontes de dados, transformando-os através do tratamento necessário para garantir a compatibilidade com os requisitos do sistema. Permite também visualizar gráficos associados aos dados, de maneira a fornecer informações relevantes para os utilizadores. Além disso, o WebSIG desenvolvido cumpre o propósito inicial, que é a implementação de um sistema capaz de monitorizar a qualidade da água em ambiente marinho, através da monitorização de diferentes variáveis e da deteção de zonas com potencial para a ocorrência algas. Esta funcionalidade de deteção de zonas com algas, está associada a níveis elevados de clorofila, que podem indicar a presença de algas. Contudo, tal como já foi referido, podem ser feitas análises de outras variáveis relevantes para este contexto, e cujos dados já existem no sistema, sendo esta uma proposta de melhoria a implementar no futuro.

Apesar dos resultados apresentados, este sistema não se encontra ainda preparado para ser colocado em produção, não podendo por isso ser aplicado ao cenário real descrito neste relatório, uma vez que é necessário efetuar mais etapas do processo de desenvolvimento, como a fase de testes. Além disso, há aspetos que podem afetar o desempenho do sistema, como o tamanho das imagens, que provocam longos tempos de espera durante o processamento das mesmas. Um exemplo disso, é a execução de ferramentas, que pode apresentar tempos de espera demasiado longos, prejudicando a experiência dos utilizadores.

Todo o trabalho desenvolvido ao longo deste projeto serviu para adquirir e consolidar os conhecimentos necessários para a concepção, planeamento, desenvolvimento e implementação de um WebSIG. Além disso, possibilitou a aprendizagem das diversas tecnologias (SIG e Web) *opensource* utilizadas ao longo do projeto.

O projeto, apesar de estar direcionado para o contexto de monitorização da água em ambiente marinho, poderá ser transformado para outros fins, através da utilização de diferentes variáveis e de diferentes fontes de dados, aumentando assim a sua versatilidade.

Bibliografia

- Ali, M. (2007). *The exploitation of remote sensing products to improve water quality monitoring in the Nile delta*.
- Anderson, D. M. (2009). Approaches to monitoring, control and management of harmful algal blooms (HABs). *Ocean & Coastal Management*, 52(7), 342–347.
<https://doi.org/10.1016/J.OCECOAMAN.2009.04.006>
- Arlinghaus, S. L., Kerski, J. J., & Arlinghaus, W. C. (2023). *Teaching Mathematics Using Interactive Mapping*. CRC Press. <https://books.google.pt/books?id=-I3hEAAAQBAJ>
- Aznar, R., Sotillo, M. G., Cailleau, S., Lorente, P., Levier, B., Amo-Baladrón, A., Reffray, G., & Álvarez-Fanjul, E. (2016). Strengths and weaknesses of the CMEMS forecasted and reanalyzed solutions for the Iberia–Biscay–Ireland (IBI) waters. *Journal of Marine Systems*, 159, 1–14.
<https://doi.org/10.1016/J.JMARSYS.2016.02.007>
- Bass, L., Clements, P., & Kazman, R. (2012). *Software Architecture in Practice: Software Architect Practice_c3*. Pearson Education. <https://books.google.pt/books?id=-II73rBDXCYC>
- Bielak, K., Borek, B., & Plechawska-Wójcik, M. (2022). Web application performance analysis using Angular, React and Vue.js frameworks. *Journal of Computer Sciences Institute*, 23, 77–83.
<https://doi.org/10.35784/jcsi.2827>
- Bierman, G., Abadi, M., & Torgersen, M. (2014). Understanding typescript. *ECOOP 2014—Object-Oriented Programming: 28th European Conference, Uppsala, Sweden, July 28–August 1, 2014. Proceedings 28*, 257–281.
- Bonocore, G., & Singhchawla, A. (2022). *Hands-On Software Architecture with Java: Learn key architectural techniques and strategies to design efficient and elegant Java applications*. Packt Publishing.
<https://books.google.pt/books?id=hWxeEAAAQBAJ>
- Boyd, C. E. (2015). *Water Quality: An Introduction*. Springer International Publishing.
<https://books.google.pt/books?id=ODwwCgAAQBAJ>
- Carmichael, W. W., & Boyer, G. L. (2016). Health impacts from cyanobacteria harmful algae blooms: Implications for the North American Great Lakes. *Harmful Algae*, 54, 194–212.
<https://doi.org/10.1016/J.HAL.2016.02.002>
- Chris K. (2021). What is HTML – Definition and Meaning of Hypertext Markup Language. 2021.
<https://www.freecodecamp.org/news/what-is-html-definition-and-meaning/>
- Ekpot W. (2021). *Angular state management made simple with NgRx*. <https://blog.logrocket.com/angular-state-management-made-simple-with-ngrx/>
- Flanagan, D. (2004). *JavaScript: o guia definitivo*. Bookman Editora.
- Foley, J., Defries, R., Asner, G., Barford, C., Bonan, G., Carpenter, S., Chapin III, F. S., Coe, M., Daily, G., Gibbs, H., Helkowski, J., Holloway, T., Howard, E., Kucharik, C., Monfreda, C., Patz, J., Prentice, I., Ramankutty, N., & Snyder, P. (2005). Global Consequences of Land Use. *Science (New York, N.Y.)*, 309, 570–574. <https://doi.org/10.1126/science.1111772>
- Fu, P. (2018). Getting to know Web GIS. *Photogrammetric Engineering & Remote Sensing*, 84(2), 59–60.

- Gackenheimer, C. (2015). *Introduction to React*. Apress.
- Gillies, J., & Cailliau, R. (2000). *How the Web was Born: The Story of the World Wide Web*. Oxford University Press. <https://books.google.pt/books?id=pIH-JijUNS0C>
- Glass, E. (2020). *How To Build a Website with CSS and HTML*. DigitalOcean. <https://books.google.pt/books?id=oh0JEAAAQBAJ>
- Goodchild, M. F., Longley, P. A., Maguire, D. J., & Rhind, D. W. (2005). Geographic information systems and science. *Wiley & Sons, West Sussex, UK*, 17, 517.
- Gourley, D., Totty, B., Sayer, M., Aggarwal, A., & Reddy, S. (2002). *HTTP: The Definitive Guide: The Definitive Guide*. O'Reilly Media. https://books.google.pt/books?id=qEoOl9bcV_cC
- Herbert D. (2023). *What is React.js? (Uses, Examples, & More)*. <https://blog.hubspot.com/website/react-js>
- Herrera, J., Carrion, D., & Brovelli, M. (2021). A COLLABORATIVE PLATFORM FOR WATER QUALITY MONITORING: SIMILE WEBGIS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLIII-B4-2021*, 201–207. <https://doi.org/10.5194/isprs-archives-XLIII-B4-2021-201-2021>
- Hillar, G. C. (2017). *MQTT Essentials - A Lightweight IoT Protocol*. Packt Publishing. <https://books.google.pt/books?id=40EwDwAAQBAJ>
- Hota, N., Dash, T., & Jain, V. (2019). *Learn Vue.js in 7 Days: Journey through Vue.js*. BPB PUBN. <https://books.google.pt/books?id=eFOwDwAAQBAJ>
- Iacovella, S. (2014). *GeoServer Cookbook*. Packt Publishing. <https://books.google.pt/books?id=oMaiBQAAQBAJ>
- Iacovella, S. (2017). *GeoServer Beginner's Guide: Share geospatial data using Open Source standards*. Packt Publishing. <https://books.google.pt/books?id=7RhKDwAAQBAJ>
- Jensen, J. R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective*. Pearson Education, Incorporated. <https://books.google.pt/books?id=IWvDrQEACAAJ>
- Jin, D. (2022). Application of remote sensing technology in water quality monitoring. *Highlights in Science, Engineering and Technology*, 17, 91–98. <https://doi.org/10.54097/hset.v17i.2511>
- Karki, S., Sultan, M., Elkadiri, R., & Elbayoumi, T. (2018). Mapping and Forecasting Onsets of Harmful Algal Blooms Using MODIS Data over Coastal Waters Surrounding Charlotte County, Florida. *Remote Sensing*, 10(10). <https://doi.org/10.3390/rs10101656>
- Li, D., & Liu, S. (2018). *Water Quality Monitoring and Management: Basis, Technology and Case Studies*. Elsevier Science. <https://books.google.pt/books?id=I2hyDwAAQBAJ>
- Longley, P. A., & Cheshire, J. A. (2017). Geographical information systems. In *The Routledge Handbook of Mapping and Cartography* (pp. 251–258). Routledge.
- McGrath, M. (2020). *HTML, CSS & JavaScript in easy steps*. In Easy Steps Limited. <https://books.google.pt/books?id=C3L1DwAAQBAJ>

- Moreira, Y. R. da S. (2022). *A expansão do cristianismo ao longo dos tempos e sua componente geoespacial expressa num WebSIG*.
- Peterson, M. P. (2012). *Online Maps with APIs and WebServices*. Springer Berlin Heidelberg.
<https://books.google.pt/books?id=9Ap61P2NiTIC>
- Petrie, C. J. (2016). *Web Service Composition*. Springer International Publishing.
<https://books.google.pt/books?id=wAlkDAAAQBAJ>
- Pollard, B. (2019). *HTTP/2 in Action*. Manning. <https://books.google.pt/books?id=azozEAAAQBAJ>
- Pramana, R., Suprpto, B. Y., & Nawawi, Z. (2021). Remote Water Quality Monitoring with Early - Warning System for Marine Aquaculture. *E3S Web of Conferences*, 324.
<https://doi.org/10.1051/e3sconf/202132405007>
- Rainer, R. K., & Prince, B. (2022). *Introduction to information systems: Supporting and transforming business*. John Wiley & Sons.
- Rajendran, K. (2022). *IMPROVING THE PERFORMANCE OF ANGULAR APPLICATION*.
- Rascia, T. (2020). *Understanding the DOM — Document Object Model*. DigitalOcean.
<https://books.google.pt/books?id=fNQBEEAAAQBAJ>
- Refsnes, H., & W3Schools. (2010). *Learn CSS with W3Schools*. Wiley.
<https://books.google.pt/books?id=opByEFgSVOoC>
- Richardson, L., & Ruby, S. (2008). *RESTful Web Services*. O'Reilly Media.
<https://books.google.pt/books?id=XUaErakHsoAC>
- Saks, E. (2019). *JavaScript Frameworks: Angular vs React vs Vue*.
- Silva, M. S. (2020). *JavaScript-Guia do Programador: Guia completo das funcionalidades de linguagem JavaScript*. Novatec Editora.
- Uzayr, S., Cloud, N., & Ambler, T. (2019). *JavaScript Frameworks for Modern Web Development: The Essential Frameworks, Libraries, and Tools to Learn Right Now*. Apress.
<https://books.google.pt/books?id=CCe7DwAAQBAJ>
- Vincent, W. S. (2022). *Django for APIs: Build web APIs with Python and Django*. WelcomeToCode.
<https://books.google.pt/books?id=0VxwDwAAQBAJ>
- Zhang, X., Zhang, Y., Qian, Z., Liu, P., Guo, R., Jin, S., Liu, J., Chen, L., Ma, Z., & Ying, L. (2020). Evaluation and Analysis of Water Quality of Marine Aquaculture Area. *International Journal of Environmental Research and Public Health*, 17, 1446. <https://doi.org/10.3390/ijerph17041446>

Apêndices

Apêndice A: Diagramas e tabelas de casos de uso elaborados pelo autor.

Apêndice B: Tabelas elaboradas pelo autor com os *endpoints* implementados no sistema gWater.

Apêndice A: Diagramas e tabelas de casos de uso

Autenticação

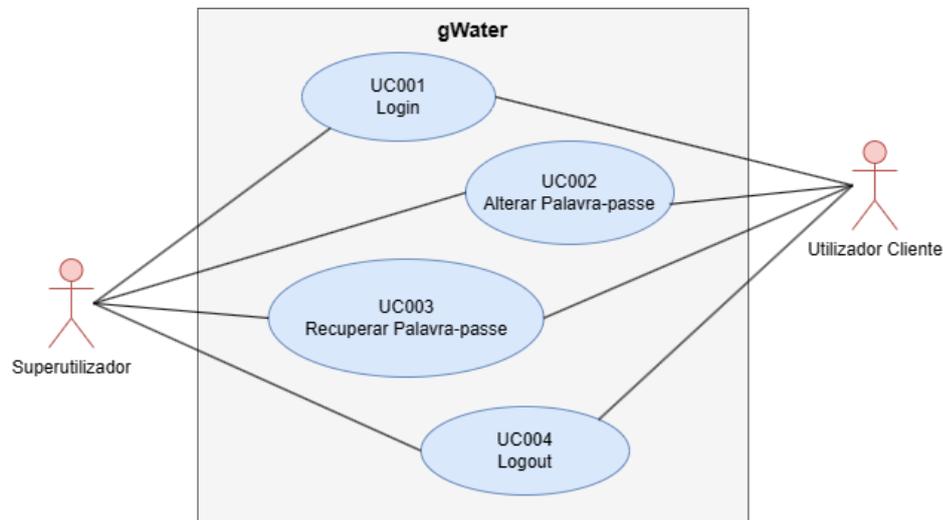


Figura 54 - Diagrama de casos de uso relacionado com a autenticação no sistema gWater.

Tabela 15 - Caso de uso "Login".

UC001 Login	
Requisitos Associados	RF001; RF002
Endpoint Associados	E_A_1
Ator	Client User, Superuser
Objetivo	O utilizador pretende autenticar-se no sistema gWater.
Pré-condições	O utilizador deve estar registado no sistema e deve saber as suas credenciais de acesso.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a página de login do sistema. 2) O utilizador introduz o <i>username</i> e <i>password</i> nos respetivos campos. 3) O utilizador clica no botão "Login". 4) O sistema confirma valida as credenciais do utilizador e devolve um <i>token</i> de acesso. 5) O utilizador é redirecionado para a página principal do sistema.
Cenário alternativo	<ol style="list-style-type: none"> 4) O utilizador insere credenciais incorretas. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem a dizer que as credenciais que foram introduzidas estão incorretas. b) O utilizador deve voltar a tentar novamente.

Tabela 16 – Caso de uso "Alterar Palavra-passe".

UC002 Alterar Palavra-passe	
Requisitos Associados	RF004
Endpoint Associado	E_A_2
Ator	Client User, Superuser
Objetivo	O utilizador pretende alterar a palavra-passe de acesso ao sistema gWater.
Pré-condições	O utilizador deve estar registado no sistema e deve estar autenticado.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a página de alteração de palavra-passe. 2) O utilizador preenche os campos necessários. 3) O utilizador clica no botão "Ok". 4) O sistema valida os dados introduzidos e faz a alteração da password. 5) O utilizador é informado do sucesso da operação.
Cenário alternativo	<ol style="list-style-type: none"> 4) O utilizador introduz palavra-passe atual incorreta ou os campos com a nova palavra-passe não coincidem. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem a informar do erro ocorrido. b) O utilizador deve voltar a tentar novamente.

Tabela 17 – Caso de uso "Recuperar Palavra-passe".

UC003 Recuperar Palavra-passe	
Requisitos Associados	RF005
Endpoint Associado	E_A_3
Ator	Client User, Superuser
Objetivo	O utilizador pretende recuperar a palavra-passe de acesso ao sistema gWater.
Pré-condições	O utilizador deve estar registado no sistema.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador clica no botão "Forgot Password". 2) O utilizador é redirecionado para a página de recuperação da palavra-passe. 3) O utilizador introduz o <i>email</i> e submete o pedido. 4) O sistema valida os dados introduzidos e envia um email para o utilizador recuperar a sua palavra-passe. 5) O utilizador é informado do sucesso da operação.
Cenário alternativo	<ol style="list-style-type: none"> 4) O utilizador introduz um email que não está registado no sistema. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem a informar do erro ocorrido. b) O utilizador deve voltar a tentar novamente.

Tabela 18 - Caso de uso "Logout".

UC004 Logout	
Requisitos Associados	RF003
Enpoint Associado	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende terminar a sessão no sistema gWater.
Pré-condições	O utilizador deve estar autenticado.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador clica no botão "Logout". 2) O sistema elimina os dados do utilizador no armazenamento do browser. 3) O utilizador é redirecionado para a página de Login.
Cenário alternativo	Não existe.

Gestão dos utilizadores

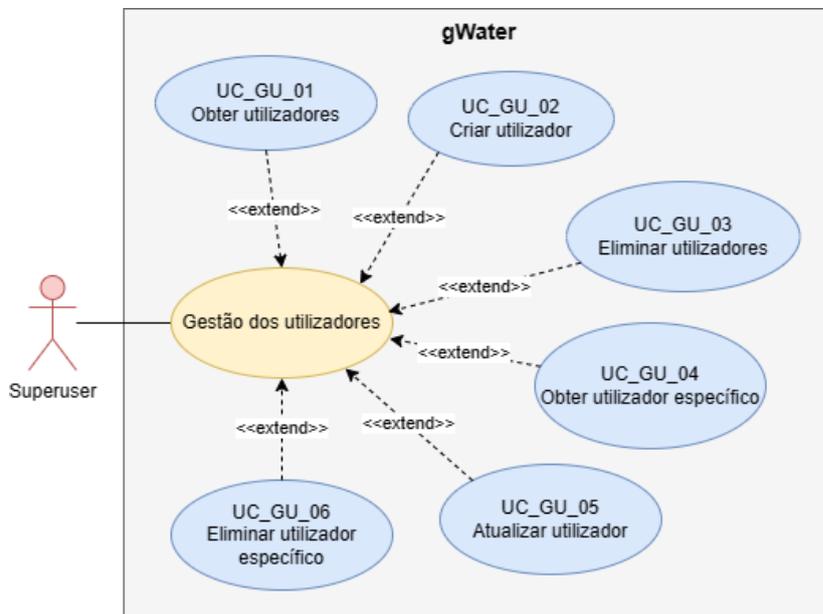


Figura 55 - Diagrama de casos de uso relacionado com a gestão dos utilizadores.

Tabela 19 – Caso de uso “Obter Utilizadores”

UC_GU_01 Obter Utilizadores	
Requisitos Associados	RF038
Endpoint Associado	E_GU_1
Ator	Superuser
Objetivo	Listar todos os utilizadores registados no sistema.
Pré-condições	O utilizador deve estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) O utilizador clica na aba relacionada com a gestão dos utilizadores. 3) O sistema apresenta uma lista de todos os utilizadores registados.
Cenário alternativo	Não existe.

Tabela 20 – Caso de uso “Criar Utilizador”

UC_GU_02 Criar Utilizador	
Requisitos Associados	RF038
Endpoint Associado	E_GU_2
Ator	Superuser
Objetivo	O utilizador pretende adicionar um Client User ao sistema gWater.
Pré-condições	O utilizador deve estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) O utilizador clica na aba relacionada com a gestão dos utilizadores. 3) O utilizador clica no botão de adicionar utilizador. 4) O utilizador preenche os campos necessários e submete o formulário. 5) O sistema valida os dados e regista um novo utilizador.
Cenário alternativo	<ol style="list-style-type: none"> 4) O utilizador não preenche todos os campos obrigatórios ou um dos campos tem dados inválidos. <ol style="list-style-type: none"> a. O sistema apresenta uma mensagem a informar do erro ocorrido. b. O utilizador deve voltar a tentar novamente.

Tabela 21 - Caso de uso “Eliminar Utilizadores”

UC_GU_03 Eliminar Utilizadores	
Requisitos Associados	RF038
Endpoint Associado	E_GU_3
Ator	Superuser
Objetivo	Eliminar todos os utilizadores do sistema
Pré-condições	O utilizador deve estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) O utilizador clica na aba relacionada com a gestão dos utilizadores. 3) O utilizador clica num utilizador específico. 4) O utilizador clica no botão para apagar o utilizador. 5) O sistema valida as permissões do autor da solicitação e remove o utilizador.
Cenário alternativo	Não existe.

Tabela 22 - Caso de uso “Obter Utilizador Específico”.

UC_GU_04 Obter Utilizador Específico	
Requisitos Associados	RF038
Endpoint Associado	E_GU_4
Ator	Superuser
Objetivo	Obter um determinado utilizador registado no sistema.
Pré-condições	O utilizador deve estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) O utilizador clica na aba relacionada com a gestão dos utilizadores. 3) O sistema apresenta uma lista de todos os utilizadores registados. 4) O utilizador escolhe um utilizador específico. 5) O sistema devolve todas as informações desse utilizador.
Cenário alternativo	Não existe.

Tabela 23 - Caso de uso “Atualizar Utilizador”.

UC_GU_05 Atualizar Utilizador	
Requisitos Associados	RF038
Endpoint Associado	E_GU_5
Ator	Superuser
Objetivo	Atualizar dados de um utilizador específico.
Pré-condições	O utilizador deve estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página dedicada atualização de dados do utilizador. 2) O utilizador preenche os dados necessários para a atualização e submete o formulário. 3) O sistema valida os dados introduzidos. 4) O sistema faz as alterações pretendidas no registo do utilizador na base de dados. 5) O sistema informa acerca do sucesso da operação.
Cenário alternativo	<ol style="list-style-type: none"> 1) O utilizador não preenche todos os campos obrigatórios ou um dos campos tem dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem a informar do erro ocorrido. b) O utilizador deve voltar a tentar novamente.

Tabela 24 - Caso de uso “Eliminar Utilizador Específico”.

UC_GU_06 Eliminar Utilizador Específico	
Requisitos Associados	RF038
Endpoint Associado	E_GU_6
Ator	Superuser
Objetivo	Eliminar um utilizador específico do sistema
Pré-condições	O utilizador deve estar autenticado e possuir as permissões necessárias.
Cenário principal	<ul style="list-style-type: none"> 6) O utilizador acede à página de gestão. 7) O utilizador clica na aba relacionada com a gestão dos utilizadores. 8) O utilizador clica num utilizador específico. 9) O utilizador clica no botão para apagar o utilizador. 10) O sistema valida as permissões do autor da solicitação e remove o utilizador.
Cenário alternativo	Não existe.

Catálogo

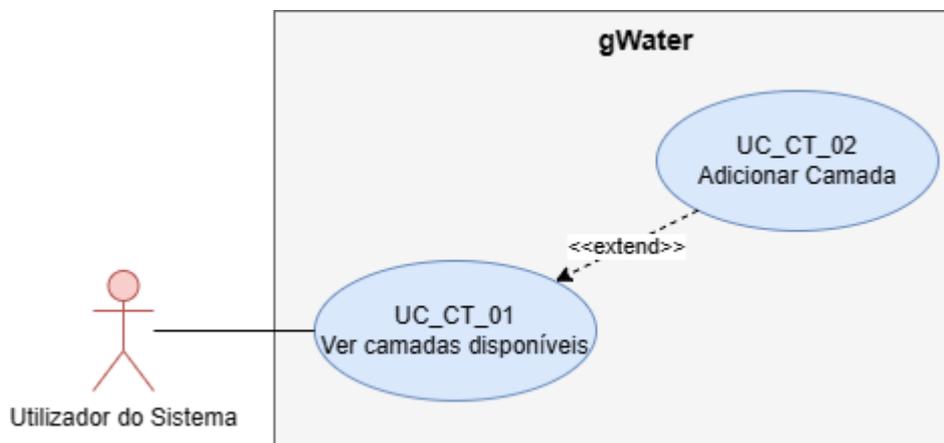


Figura 56 - Diagrama de casos de uso relacionado com o catálogo de dados do sistema gWater.

Tabela 25 - Caso de uso "Visualizar camadas disponíveis".

UC_CT_01 Visualizar camadas disponíveis	
Requisitos Associados	RF006; RF007; RF008
Endpoint Associado	E_GU_4
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar uma lista das camadas disponíveis no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega na estrutura em árvore do catálogo de dados do sistema e seleciona uma camada. 2) A camada é adicionada à propriedade "layers" de um subestado do estado principal da aplicação. 3) A camada fica disponível no Map Layers, para criação de gráficos e para ser utilizada como input na execução de ferramentas.
Cenário alternativo	Não existe.

Tabela 26 - Caso de uso "Adicionar Camada ao Map Layers".

UC_CT_02 Adicionar Camada ao Map Layers	
Requisitos Associados	RF010
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende adicionar uma camada presente no catálogo de dados ao Map Layers, com o objetivo de utilizar essa camada para a criação de gráficos, utilização de ferramentas ou simplesmente visualizá-la no mapa.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega na estrutura em árvore do catálogo de dados do sistema e seleciona uma camada. 2) A camada é adicionada à propriedade “layers” de um subestado do estado principal da aplicação (<i>Store</i> NgRx). 3) A camada fica disponível no Map Layers, para criação de gráficos e para ser utilizada como input na execução de ferramentas.
Cenário alternativo	Não existe.

Map Layers

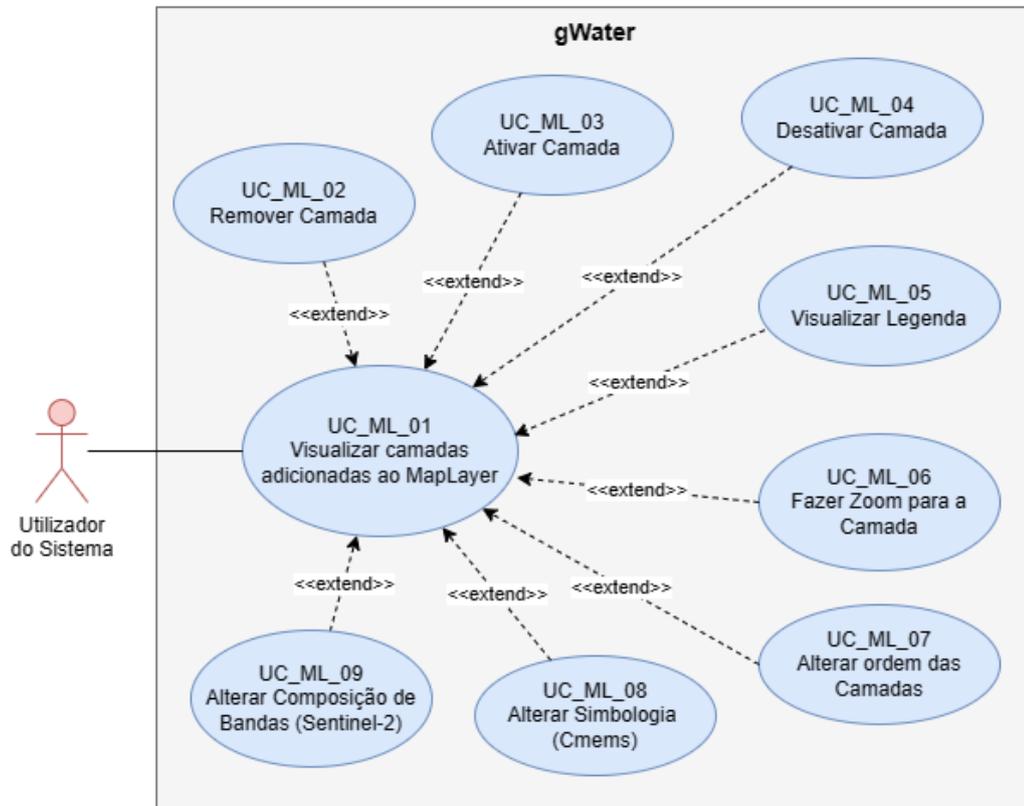


Figura 57 - Diagrama de casos de uso associado às operações do Map Layers.

Tabela 27 - Caso de uso "Visualizar Camadas do Map Layers".

UC_ML_01 Visualizar Camadas do Map Layers	
Requisitos Associados	-
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar as camadas que foram adicionadas ao Map Layers através do catálogo. Essas camadas estão na aba Map Layers.
Pré-condições	O utilizador tem de estar autenticado.
Cenário principal	1) O utilizador adiciona camadas ao Map Layers através do catálogo. 2) Navega para a aba Map Layers e visualiza as camadas disponíveis para realizar diversas operações.
Cenário alternativo	Não existe.

Tabela 28 - Caso de uso "Remover Camada do Map Layers".

UC_ML_02 Remover Camada do Map Layers	
Requisitos Associados	RF011
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende remover uma camada presente no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e a camada tem de estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 3) O utilizador navega para a lista de camadas presentes no Map Layers. 4) Clica no ícone de opções da camada e escolhe a opção "Remove". 5) A camada é removida do Map Layers e deixa de estar disponível para visualização no mapa, criação de gráficos e execução de ferramentas.
Cenário alternativo	Não existe.

Tabela 29 - Caso de uso "Ativar Camada do Map Layers".

UC_ML_03 Ativar Camada do Map Layers	
Requisitos Associados	RF012
Endpoints Associados	E_MP_1; E_MP_2
Ator	Client User, Superuser
Objetivo	O utilizador pretende ativar uma camada presente no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e a camada tem de estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica caixa de seleção da camada. 3) A camada é ativada e surge no mapa da página principal.
Cenário alternativo	Não existe.

Tabela 30 - Caso de uso "Desativar Camada do Map Layers".

UC_ML_04 Desativar Camada do Map Layers	
Requisitos Associados	RF013
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende desativar uma camada presente no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e a camada tem de estar presente no Map Layers e ativada.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica caixa de seleção da camada e esta fica desmarcada. 3) A camada é desativada e é removida do mapa da página principal.
Cenário alternativo	Não existe.

Tabela 31 - Caso de uso "Visualizar Legenda".

UC_ML_05 Visualizar Legenda	
Requisitos Associados	RF022
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar a legenda de uma camada presente no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e a camada tem de estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica na seta de expansão da camada. 3) Surge a legenda por baixo do item da camada no Map Layers.
Cenário alternativo	Não existe.

Tabela 32 - Caso de uso "Fazer Zoom para a Camada do Map Layers".

UC_ML_06 Fazer Zoom para a camada do Map Layers	
Requisitos Associados	RF040
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende alterar a ordem das camadas no Map Layers, com influência na visualização no mapa.
Pré-condições	O utilizador tem de estar autenticado e devem existir pelo menos duas camadas no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica nas opções da camada e clica em "Zoom to Layer". 3) O mapa faz zoom para a camada especificada.
Cenário alternativo	Não existe.

Tabela 33 - Caso de uso "Alterar Ordem das Camadas no Map Layers".

UC_ML_07 Alterar Ordem das Camadas no Map Layers	
Requisitos Associados	RF024
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende alterar a ordem das camadas no Map Layers, com influência na visualização no mapa.
Pré-condições	O utilizador tem de estar autenticado e devem existir pelo menos duas camadas no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica numa camada e mantém premido enquanto arrasta a camada para a posição pretendida. 3) A camada muda de posição na lista de camadas e caso esteja ativa, essa mudança é refletida no mapa.
Cenário alternativo	Não existe.

Tabela 34 - Caso de uso "Alterar a Simbologia das Camadas do CMEMS".

UC_ML_08 Alterar a Simbologia das Camadas do CMEMS	
Requisitos Associados	RF023
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende alterar a simbologia de determinada camada CMEMS presente no mapa.
Pré-condições	O utilizador tem de estar autenticado e a camada deve estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica no ícone de opções da camada e escolhe a opção "Change Symbology". 3) Escolhe a simbologia desejada. 4) A simbologia é alterar e é refletida na legenda e no mapa.
Cenário alternativo	Não existe.

Tabela 35 - Caso de uso "Alterar a Composição das Bandas das Cmadas Sentinel-2".

UC_ML_09 Alterar a Composição das Bandas das camadas Sentinel-2	
Requisitos Associados	RF026
Endpoints Associados	-
Ator	Client User, Superuser
Objetivo	O utilizador pretende alterar a composição das bandas de determinada camada Sentinel-2 presente no mapa.
Pré-condições	O utilizador tem de estar autenticado e a camada deve estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a lista de camadas presentes no Map Layers. 2) Clica no ícone de opções da camada e escolhe a opção "Change Symbology". 3) Escolhe a composição de bandas desejada. 4) A simbologia é alterar e é refletida na legenda e no mapa.
Cenário alternativo	Não existe.

Gráficos

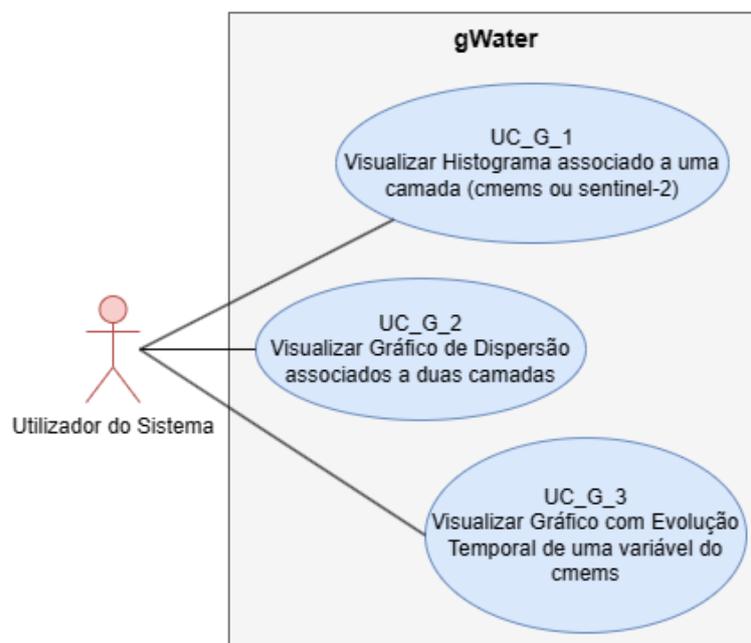


Figura 58 - Diagrama de casos de uso relacionado com a apresentação de gráficos no sistema gWater.

Tabela 36 - Caso de uso "Visualizar Histograma associado a uma camada".

UC_G_1 Visualizar Histograma	
Requisitos Associados	RF014; RF015; RF016
Endpoints Associados	E_G_1
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar um histograma associado a determinada camada CMEMS ou Sentinel-2 presente no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e a camada deve estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a opção "Charts" no menu do lado direito da página principal. 2) Clica na opção "Histogram" nas abas que contém os gráficos disponíveis. 3) Escolhe a camada pretendida e insere outros dados necessários. 4) O sistema apresenta um histograma conforme os dados fornecidos pelo utilizador.
Cenário alternativo	Não existe

Tabela 37 - Caso de uso "Visualizar Gráfico de Dispersão".

UC_G_2 Visualizar Gráfico de Dispersão	
Requisitos Associados	RF017; RF018; RF019; RF020
Endpoints Associados	E_G_2
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar um gráfico de dispersão que mostra a relação entre duas camadas do CMEMS presentes no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e as camadas deve estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a opção "Charts" no menu do lado direito da página principal. 2) Clica na opção "Scatter Plot" nas abas que contém os gráficos disponíveis. 3) Escolhe as camadas do CMEMS pretendidas. 4) Faz upload de um ficheiro em formato <i>zip</i> que contém um <i>shapefile</i> com os pontos pretendidos para a criação do gráfico. 5) O sistema apresenta um gráfico de dispersão conforme os dados fornecidos pelo utilizador.
Cenário alternativo	<ol style="list-style-type: none"> 3) O utilizador introduz camadas de diferentes áreas de interesse. <ol style="list-style-type: none"> a) O sistema informa que as camadas têm de ser da mesma área de interesse. 4) O utilizador marca os pontos pretendidos no mapa.

Tabela 38 - Caso de uso "Visualizar Gráfico de Evolução Temporal".

UC_G_3 Visualizar Gráfico de Evolução Temporal	
Requisitos Associados	RF021
Endpoints Associados	E_G_3
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar um gráfico de evolução temporal associado a determinada camada CMEMS presente no Map Layers.
Pré-condições	O utilizador tem de estar autenticado e a camada deve estar presente no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a opção "Charts" no menu do lado direito da página principal. 2) Clica na opção "Temporal Evolution" nas abas que contém os gráficos disponíveis. 3) Escolhe a camada pretendida (CMEMS). 4) Faz upload de um ficheiro em formato <i>zip</i> que contém um <i>shapefile</i> com um ponto que será utilizado para obter os valores de cada imagem de diferente data nesse ponto. 5) O sistema apresenta um gráfico de evolução temporal conforme os dados fornecidos pelo utilizador.
Cenário alternativo	Não existe

Ferramentas

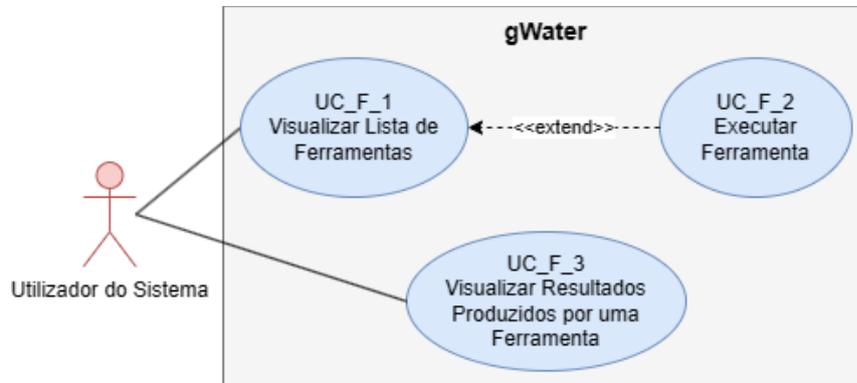


Figura 59 - Diagrama de casos de uso relacionado com a execução de ferramentas no sistema gWater.

Tabela 39 - Caso de uso "Visualizar Lista de Ferramentas".

UC_F_1 Visualizar Lista de Ferramentas	
Requisitos Associados	RF027
Endpoints Associados	E_F_1
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar navegar na lista de ferramentas disponíveis no sistema.
Pré-condições	O utilizador tem de estar autenticado.
Cenário principal	1) O utilizador navega para a opção "Tools" no menu do lado direito da página principal. 2) O utilizador visualiza navega na estrutura em árvore da lista de ferramentas existentes no sistema.
Cenário alternativo	Não existe.

Tabela 40 - Caso de uso "Executar Ferramenta".

UC_F_2 Executar Ferramenta	
Requisitos Associados	RF028; RF029; RF030; RF031
Endpoints Associados	E_RT_2; E_RR_2
Ator	Client User, Superuser
Objetivo	O utilizador pretende executar uma ferramenta do sistema.
Pré-condições	O utilizador tem de estar autenticado e as camadas que pretende utilizar como input devem estar presentes no Map Layers.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega para a opção "Tools" no menu do lado direito da página principal. 2) O utilizador navega na estrutura em árvore da lista de ferramentas existentes no sistema e clica na ferramenta pretendida. 3) Insere os dados necessários e submete o formulário ao clicar no botão "Run". 4) O utilizador deverá poder continuar a utilizar o sistema normalmente sendo notificado ao longo do processo de execução da ferramenta. 5) O resultado ou resultados da execução da ferramenta são adicionados ao catálogo dos dados, podendo ser adicionados ao Map Layers pelo utilizador.
Cenário alternativo	<ol style="list-style-type: none"> 3) O utilizador introduz dados incorretos. <ol style="list-style-type: none"> a) O sistema informa o utilizador acerca dos erros encontrados.

Tabela 41 - Caso de uso "Visualizar Resultados Produzidos pelas Ferramentas".

UC_F_3 Visualizar Resultados Produzidos Pelas Ferramentas	
Requisitos Associados	RF032
Endpoints Associados	E_RR_1
Ator	Client User, Superuser
Objetivo	O utilizador pretende visualizar os dados produzidos pela execução de uma ferramenta.
Pré-condições	O utilizador tem de estar autenticado e a execução da ferramenta tem de estar concluída.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador navega na estrutura em árvore do catálogo dos dados existentes no sistema e clica em "Tool Outputs" dentro da respetiva área de interesse. 2) O utilizador adiciona uma ou mais camadas relacionadas com resultados de ferramentas ao Map Layers. 3) A camada fica disponível no Map Layers.
Cenário alternativo	Não existe.

Gestão dos Estilos das Camadas CMEMS

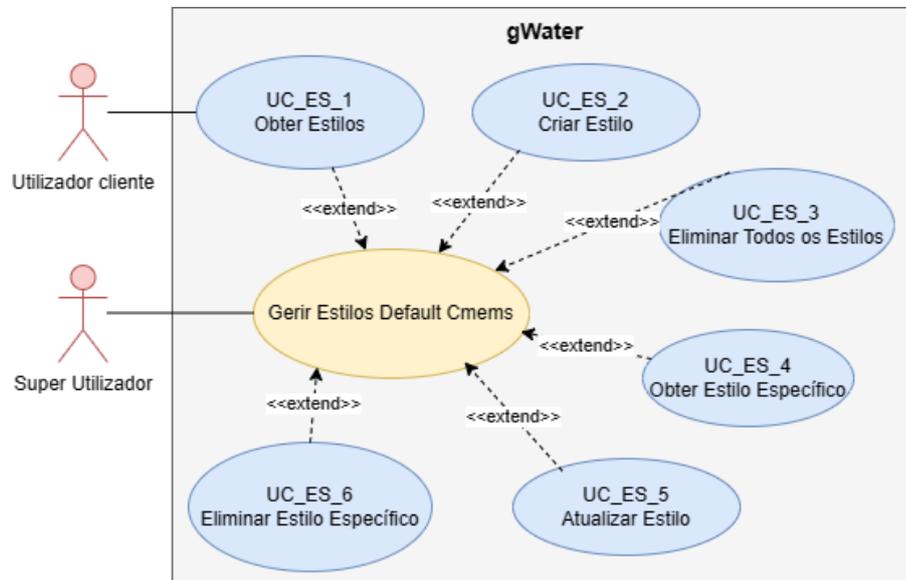


Figura 60 - Diagrama de casos de uso relacionado com a gestão dos estilos das Camadas CMEMS.

Tabela 42 - Caso de uso "Visualizar Estilos CMEMS".

UC_ES_1 Visualizar Estilos CMEMS	
Requisitos Associados	RF039
Endpoints Associados	E_ES_1
Ator	Superuser, Client User
Objetivo	O Super User pretende visualizar os estilos associados às camadas CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal (Superuser, Client User)	1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos. 3) Visualiza todos os estilos existentes no sistema.
	1) O utilizador acede à página principal do sistema. 2) O utilizador clica na opção que permite alterar a simbologia de uma camada do CMEMS, no Map Layers. 3) Visualiza uma lista com os estilos disponíveis.
Cenário alternativo	Não existe

Tabela 43 - Caso de uso "Criar Estilo CMEMS".

UC_ES_2 Criar Estilo CMEMS	
Requisitos Associados	RF039
Endpoints Associados	E_ES_2
Ator	Superuser
Objetivo	O utilizador pretende criar um estilo no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos. 3) Preenche os campos necessários e submete o formulário. 4) O novo estilo é adicionado à base de dados 5) O sistema notifica o utilizador acerca do sucesso da operação.
Cenário alternativo	<ol style="list-style-type: none"> 3) O utilizador fornece dados incorretos no formulário. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem de erro.

Tabela 44 - Caso de uso "Eliminar Todos os Estilos das Camadas CMEMS".

UC_ES_3 Eliminar Todos os Estilos CMEMS	
Requisitos Associados	RF039
Endpoints Associados	E_ES_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todos os estilos CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos. 3) Clica no botão para eliminar todos os estilos.
Cenário alternativo	Não existe.

Tabela 45 - Caso de uso "Obter Estilo Específico das Camadas CMEMS".

UC_ES_4 Obter Estilo Específico CMEMS	
Requisitos Associados	RF039
Endpoints Associados	E_ES_4
Ator	Superuser
Objetivo	O utilizador pretende obter um estilo específico do sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos. 3) Visualiza uma lista com todos os estilos existentes no sistema. 4) Seleciona um dos estilos da lista ou pesquisa o estilo no campo de procura. 5) O sistema devolve todas as informações sobre o estilo escolhido.
Cenário alternativo	Não existe.

Tabela 46 - Caso de uso "Editar Estilo das Camadas CMEMS".

UC_ES_5 Editar Estilo CMEMS	
Requisitos Associados	RF039
Endpoints Associados	E_ES_5
Ator	Superuser
Objetivo	O utilizador pretende editar um estilo existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos. 3) Seleciona o estilo que pretende editar. 4) Introduce os dados necessários e submete o formulário. 5) O sistema notifica o utilizador acerca do sucesso da operação.
Cenário alternativo	<ol style="list-style-type: none"> 4) O utilizador introduz dados incorretos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem de erro.

Tabela 47 - Caso de uso "Remover Estilo das Camadas CMEMS".

UC_ES_6 Remover Estilo	
Requisitos Associados	RF039
Endpoints Associados	E_ES_6
Ator	Superuser
Objetivo	O utilizador pretende remover um estilo no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos. 3) Seleciona o estilo que pretende remover. 4) Clica no botão para remover o estilo do sistema. 5) O sistema notifica o utilizador acerca do sucesso da operação.
Cenário alternativo	Não existe.

Gestão dos Estilos do GeoServer

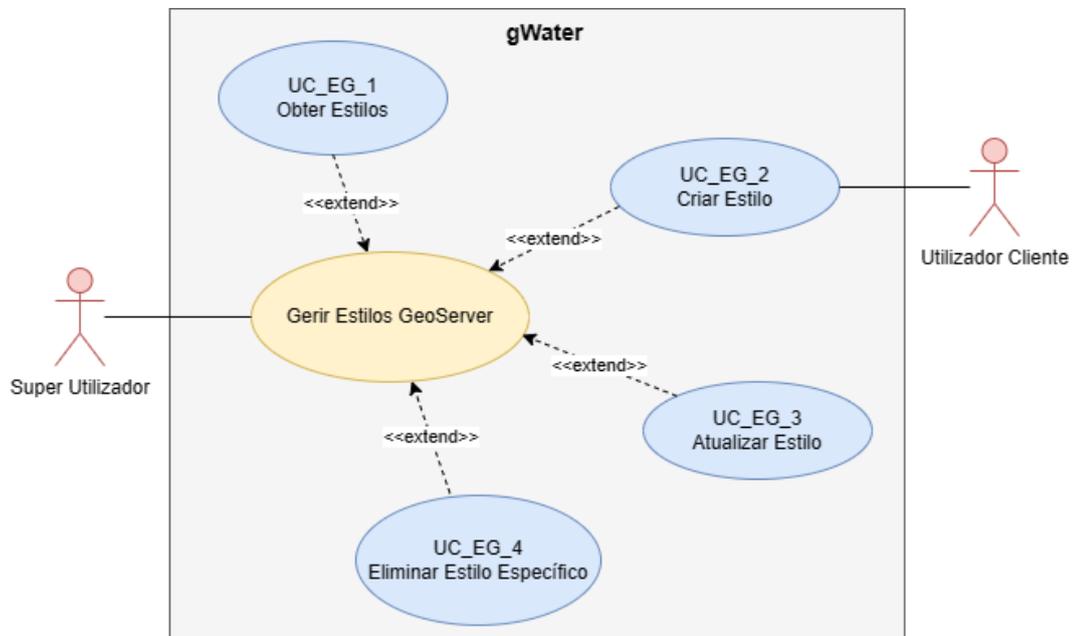


Figura 61 - Diagrama de casos de uso relacionados com a gestão de estilo no GeoServer.

Tabela 48 - Caso de uso "Obter Estilos do GeoServer".

UC_EG_1 Obter Estilos GeoServer	
Requisitos Associados	RF046
Endpoints Associados	E_EG_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar os estilos existentes no GeoServer.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos do GeoServer. 3) Visualiza todos os estilos existentes no GeoServer.
Cenário alternativo	Não existe

Tabela 49 - Caso de uso "Criar Estilo do GeoServer".

UC_EG_2 Criar Estilo GeoServer	
Requisitos Associados	RF046
Endpoints Associados	E_ES_2
Ator	Superuser, Client User
Objetivo	O utilizador pretende criar um estilo no GeoServer.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal (Superuser, Client User)	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos do GeoServer. 3) Clica em adicionar estilo. 4) Preenche os campos necessários. 5) O sistema apresenta uma mensagem de sucesso.
	<ol style="list-style-type: none"> 4) O utilizador acede à página principal do sistema. 5) O utilizador clica na opção que permite alterar a simbologia de uma camada do CMEMS, no Map Layers. 6) Visualiza uma lista com os estilos CMEMS disponíveis. 7) Escolhe o estilo pretendido e clica "OK". 8) É criado um estilo no GeoServer com os dados fornecidos pelo utilizador. 9) O sistema apresenta uma mensagem de sucesso.
Cenário alternativo (Superuser, Client User)	<ol style="list-style-type: none"> 4) O utilizador introduz valores inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem de erro.
	<p>Não existe uma vez que só pode escolher entre os valores fornecidos pelo sistema.</p>

Tabela 50 - Caso de uso "Editar Estilo do GeoServer".

UC_EG_3 Editar Estilo GeoServer	
Requisitos Associados	RF046
Endpoints Associados	E_EG_3
Ator	Superuser
Objetivo	O utilizador pretende editar um estilo existente no GeoServer.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos do GeoServer. 3) Visualiza todos os estilos existentes no GeoServer. 4) Escolhe um dos estilos da lista ou pesquisa através do campo de pesquisa. 5) Atualiza os dados de um estilo específico e submete o formulário. 6) O sistema apresenta uma mensagem de sucesso.
Cenário alternativo	<ol style="list-style-type: none"> 5) O utilizador introduz dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem de erro.

Tabela 51 - Caso de uso "Eliminar Estilo do GeoServer".

UC_EG_4 Eliminar Estilo GeoServer	
Requisitos Associados	RF046
Endpoints Associados	E_EG_4
Ator	Superuser
Objetivo	O utilizador pretende eliminar um estilo existente no GeoServer.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Seleciona gestão dos estilos do GeoServer. 3) Visualiza todos os estilos existentes no GeoServer. 4) Escolhe um dos estilos da lista ou pesquisa através do campo de pesquisa. 5) Clica no botão para eliminar o estilo. 6) O sistema apresenta uma mensagem de sucesso.
Cenário alternativo	Não existe.

Gestão dos *Workspaces* do GeoServer

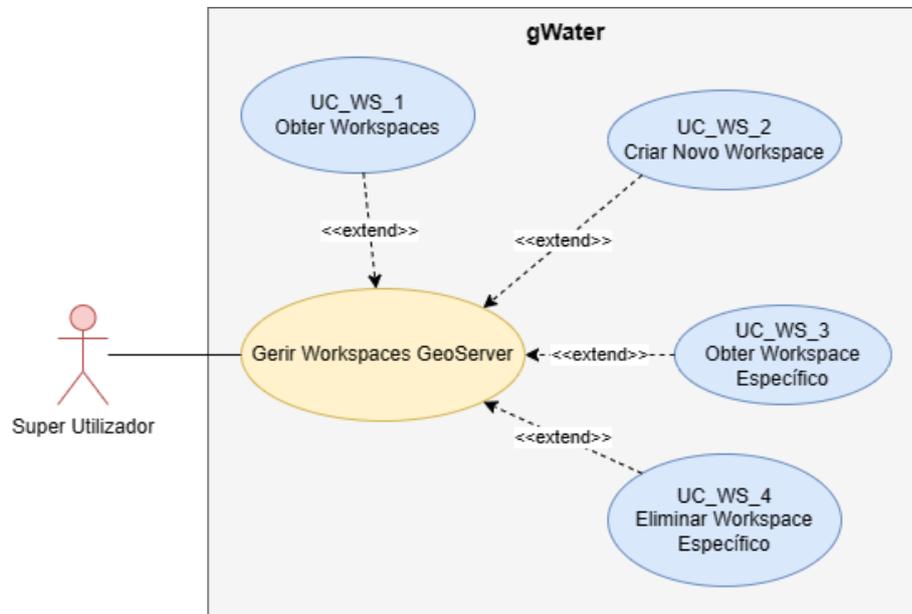


Figura 62 - Diagrama de casos de uso relacionado com a gestão dos workspaces do GeoServer.

Tabela 52 - Caso de uso "Obter Workspaces".

UC_WS_1 Obter <i>Workspaces</i>	
Requisitos Associados	RF043
Endpoints Associados	E_WS_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma lista com todos os <i>Workspaces</i> existentes no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão dos <i>workspaces</i> . 4) Visualiza a lista com todos os <i>workspaces</i> existentes.
Cenários alternativo	Não existe.

Tabela 53 - Caso de uso "Criar *Workspace*".

UC_WS_2 Criar <i>Workspace</i>	
Requisitos Associados	RF043
Endpoints Associados	E_WS_2
Ator	Superuser
Objetivo	O utilizador pretende criar um <i>Workspace</i> no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão dos <i>workspaces</i>. 4) Clica em adicionar um <i>workspace</i>. 5) Fornece os dados necessários e submete o pedido. 6) O sistema valida os dados, cria um <i>workspace</i> e apresenta uma mensagem de sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 6) Os dados submetidos são inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem a informar acerca do erro ocorrido.

Tabela 54 - Caso de uso "Obter *Workspace* Específico".

UC_WS_3 Obter <i>Workspace</i> Específico	
Requisitos Associados	RF043
Endpoints Associados	E_WS_3
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as informações relacionadas com um <i>Workspace</i> específico existente no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão dos <i>workspaces</i>. 4) Visualiza a lista com todos os <i>workspaces</i> existentes. 5) Clica num <i>workspace</i> específico para visualizar todas as informações relacionadas com o mesmo.
Cenários alternativo	Não existe,

Tabela 55 - Caso de uso "Eliminar *Workspace*".

UC_WS_4 Eliminar <i>Workspace</i>	
Requisitos Associados	RF043
Endpoints Associados	E_WS_4
Ator	Superuser
Objetivo	O utilizador pretende eliminar um <i>Workspace</i> no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão dos <i>workspaces</i>. 4) Escolhe um <i>workspace</i> na lista de <i>workspaces</i> existentes no sistema. 5) Clica no botão para eliminar o <i>workspace</i>. 6) O sistema pergunta ao utilizador se deseja prosseguir com a operação. 7) O utilizador responde que sim. 8) O sistema executa o pedido e apresenta uma mensagem a informar do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> b) O sistema não efetua a operação.

Gestão das Stores do GeoServer

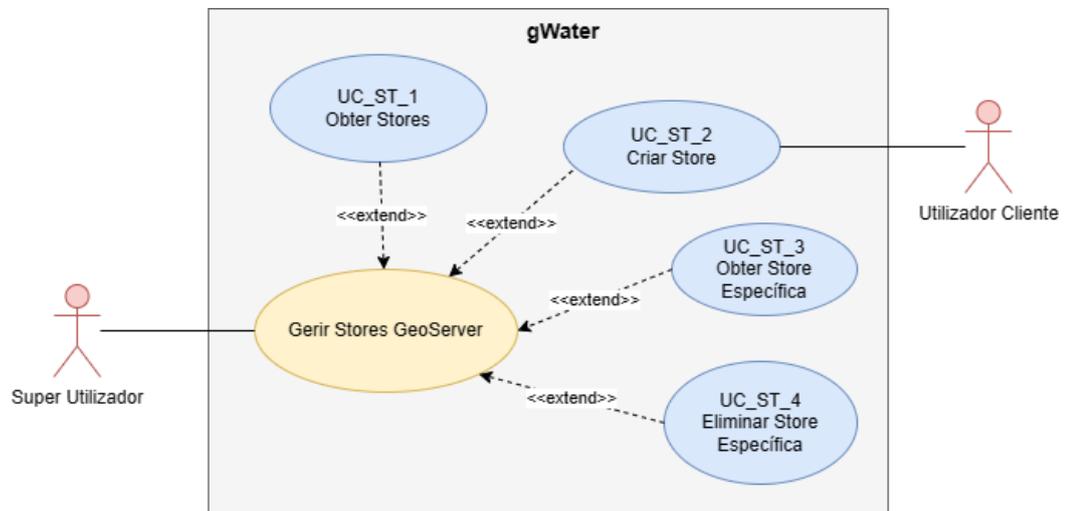


Figura 63 - Diagrama de casos de uso relacionado com a gestão das stores do GeoServer.

Tabela 56 - Caso de uso "Obter Stores".

UC_ST_1 Obter Stores	
Requisitos Associados	RF044
Endpoints Associados	E_ST_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma lista com todas as <i>Stores</i> existentes no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das <i>Stores</i>. 4) Visualiza a lista com todas as <i>Stores</i> existentes.
Cenários alternativo	Não existe.

Tabela 57 - Caso de uso "Criar *Store*".

UC_ST_2 Criar <i>Store</i>	
Requisitos Associados	RF044
Endpoints Associados	E_ST_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma <i>store</i> no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das <i>stores</i>. 4) Clica em adicionar uma <i>store</i>. 5) Fornece os dados necessários e submete o pedido. 6) O sistema valida os dados, cria uma <i>store</i> e apresenta uma mensagem de sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 6) Os dados submetidos são inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 58 - Caso de uso "Obter *Store* Específica".

UC_ST_3 Obter <i>Store</i> Específica	
Requisitos Associados	RF044
Endpoints Associados	E_ST_3
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as informações relacionadas com uma <i>store</i> específica existente no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das <i>stores</i>. 4) Visualiza a lista com todas as <i>stores</i> existentes. 5) Clica numa <i>store</i> específica para visualizar todas as informações relacionadas com a mesma.
Cenários alternativo	Não existe.

Tabela 59 - Caso de uso "Eliminar *Store*".

UC_ST_4 Eliminar <i>Store</i>	
Requisitos Associados	RF044
Endpoints Associados	E_ST_4
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma <i>store</i> no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das <i>Stores</i>. 4) Escolhe uma <i>Store</i> na lista de <i>Stores</i> existentes no sistema. 5) Clica no botão para eliminar a <i>Store</i>. 6) O sistema pergunta ao utilizador se deseja prosseguir com a operação. 7) O utilizador responde que sim. 8) O sistema executa o pedido e apresenta uma mensagem a informar do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Camadas do GeoServer

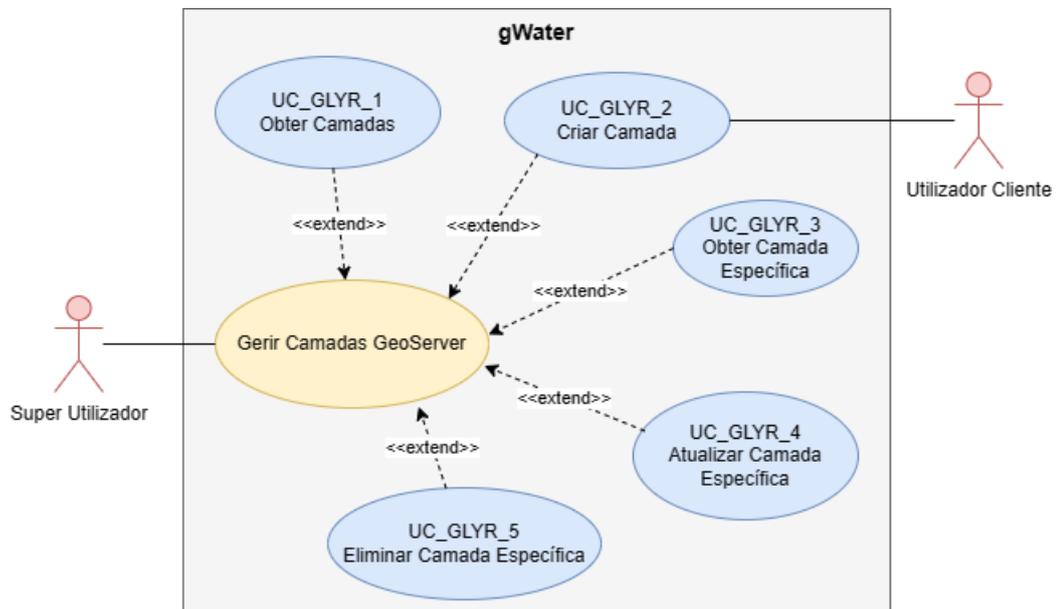


Figura 64 - Diagrama de casos de uso relacionado com a gestão das camadas do GeoServer.

Tabela 60 - Caso de uso "Obter Camadas do GeoServer".

UC_GLYR_1 Obter Camadas GeoServer	
Requisitos Associados	RF045
Endpoints Associados	E_GLYR_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma lista com todos as camadas existentes no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das camadas. 4) Visualiza a lista com todos as camadas existentes.
Cenários alternativo	Não existe.

Tabela 61 - Caso de uso "Criar Camada do GeoServer".

UC_GLYR_2 Criar Camada GeoServer	
Requisitos Associados	RF045
Endpoints Associados	E_GLYR_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma camada no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das camadas. 4) Clica em adicionar uma camada. 5) Fornece os dados necessários e submete o pedido. 6) O sistema valida os dados, cria uma camada e apresenta uma mensagem de sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 6) Os dados submetidos são inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 62 - Caso de uso "Obter Camada Específica do GeoServer".

UC_GLYR_3 Obter Camada GeoServer Específica	
Requisitos Associados	RF045
Endpoints Associados	E_GLYR_3
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as informações relacionadas com uma camada específica existente no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das camadas. 4) Visualiza a lista com todas as camadas existentes. 5) Clica numa camada específica para visualizar todas as informações relacionadas com a mesma.
Cenários alternativo	Não existe.

Tabela 63 - Caso de uso "Atualizar Camada do GeoServer".

UC_GLYR_4 Atualizar Camada GeoServer	
Requisitos Associados	RF045
Endpoints Associados	E_GLYR_4
Ator	Superuser
Objetivo	O utilizador pretende atualizar as informações relacionadas com uma camada específica existente no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das camadas. 4) Escolhe uma das camadas da lista ou pesquisa através do campo de pesquisa. 5) Preenche e submete o formulário. 6) O sistema valida e apresenta uma mensagem de sucesso.
Cenários alternativo	<ol style="list-style-type: none"> 6) O utilizador introduz dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem de erro.

Tabela 64 - Caso de uso "Eliminar Camada do GeoServer".

UC_GLYR_5 Eliminar Camada do GeoServer	
Requisitos Associados	RF045
Endpoints Associados	E_GLYR_5
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma camada no Geoserver.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão do GeoServer. 3) Clica em gestão das camadas. 4) Escolhe uma camada na lista de camadas existentes no sistema. 5) Clica no botão para eliminar a camada. 6) O sistema pergunta ao utilizador se deseja prosseguir com a operação. 7) O utilizador responde que sim. 8) O sistema executa o pedido e apresenta uma mensagem a informar do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Variáveis

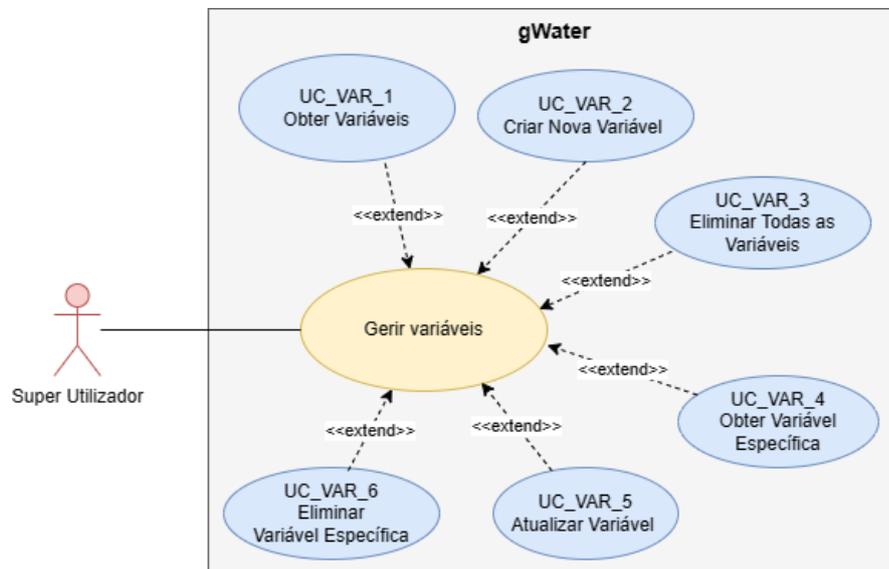


Figura 65 - Diagrama de casos de uso relacionado com a gestão das variáveis do sistema gWater.

Tabela 65 - Caso de uso "Obter Variáveis".

UC_VAR_1 Obter Variáveis	
Requisitos Associados	RF041
Endpoints Associados	E_VAR_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as variáveis existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis. 3) Visualiza a lista completa de variáveis existentes no sistema.
Cenários alternativo	Não existe.

Tabela 66 - Caso de uso "Criar Variável".

UC_VAR_2 Criar Variável	
Requisitos Associados	RF041
Endpoints Associados	E_VAR_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma variável no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis. 3) Clica em adicionar variável. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma variável. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 67 - Caso de uso "Eliminar Variáveis".

UC_VAR_3 Eliminar Variáveis	
Requisitos Associados	RF041
Endpoints Associados	E_VAR_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as variáveis existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis. 3) Clica na opção para eliminar todas as variáveis. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as variáveis e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 68 - Caso de uso "Obter Variável Específica".

UC_VAR_4 Obter Variável Específica	
Requisitos Associados	RF041
Endpoints Associados	E_VAR_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma variável específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis. 3) Escolha uma variável na lista completa de variáveis. 4) O sistema devolve todas as informações relacionadas com essa variável.
Cenários alternativo	Não existe.

Tabela 69 - Caso de uso "Editar Variável".

UC_VAR_5 Editar Variável	
Requisitos Associados	RF041
Endpoints Associados	E_VAR_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma variável no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis. 3) Clica numa variável específica na lista de variáveis existentes no sistema. 4) Clica no botão para editar a variável. 5) Fornece todos os dados necessários e submete o formulário. 6) O sistema valida os dados introduzidos e atualiza a variável. 7) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 6) O utilizador introduziu dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 70 - Caso de uso "Eliminar Variável Específica".

UC_VAR_6 Eliminar Variável Específica	
Requisitos Associados	RF041
Endpoints Associados	E_VAR_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma variável específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis. 3) Escolha uma variável na lista completa de variáveis. 4) O sistema devolve todas as informações relacionadas com essa variável. 5) Clica na opção para eliminar a variável. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a variável e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Variáveis CMEMS

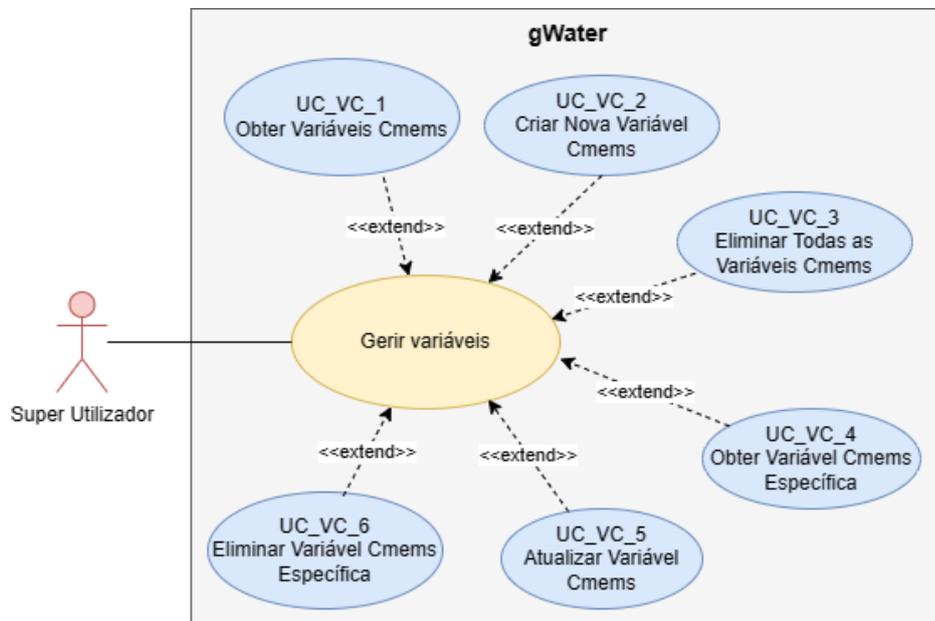


Figura 66 - Diagrama de casos de uso relacionado com a gestão das variáveis CMEMS.

Tabela 71 - Caso de uso "Obter Variáveis CMEMS".

UC_VC_1 Obter Variáveis CMEMS	
Requisitos Associados	RF041
Endpoints Associados	E_VC_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as variáveis CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	4) O utilizador acede à página de gestão. 5) Accede à secção relacionada com a gestão das variáveis CMEMS. 6) Visualiza a lista completa de variáveis CMEMS existentes no sistema.
Cenários alternativo	Não existe.

Tabela 72 - Caso de uso "Criar Variável CMEMS".

UC_VC_2 Criar Variável CMEMS	
Requisitos Associados	RF041
Endpoints Associados	E_VC_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma variável CMEMS no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis CMEMS. 3) Clica em adicionar variável CMEMS. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma variável CMEMS. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 73 - Caso de uso "Eliminar Variáveis CMEMS".

UC_VC_3 Eliminar Variáveis CMEMS	
Requisitos Associados	RF041
Endpoints Associados	E_VC_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as variáveis CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis CMEMS. 3) Clica na opção para eliminar todas as variáveis CMEMS. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as variáveis CMEMS e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 74 - Caso de uso "Obter Variável CMEMS Específica".

UC_VC_4 Obter Variável CMEMS Específica	
Requisitos Associados	RF041
Endpoints Associados	E_VC_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma variável CMEMS específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das variáveis CMEMS. 3) Escolha uma variável CMEMS na lista completa de variáveis. 4) O sistema devolve todas as informações relacionadas com essa variável CMEMS.
Cenários alternativo	Não existe.

Tabela 75 - Caso de uso "Editar Variável CMEMS".

UC_VC_5 Editar Variável CMEMS	
Requisitos Associados	RF041
Endpoints Associados	E_VC_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma variável CMEMS no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das variáveis CMEMS. 3) Clica numa variável específica na lista de variáveis existentes no sistema. 4) Clica no botão para editar a variável CMEMS. 5) Fornece todos os dados necessários e submete o formulário. 6) O sistema valida os dados introduzidos e atualiza a variável CMEMS. 7) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 6) O utilizador introduziu dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 76 - Caso de uso "Eliminar Variável CMEMS Específica".

UC_VC_6 Eliminar Variável CMEMS Específica	
Requisitos Associados	RF041
Endpoints Associados	E_VC_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma variável CMEMS específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das variáveis CMEMS. 3) Escolha uma variável CMEMS na lista completa de variáveis CMEMS. 4) O sistema devolve todas as informações relacionadas com essa variável CMEMS. 5) Clica na opção para eliminar a variável CMEMS. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a variável CMEMS e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão dos Mosaicos do Sentinel-2

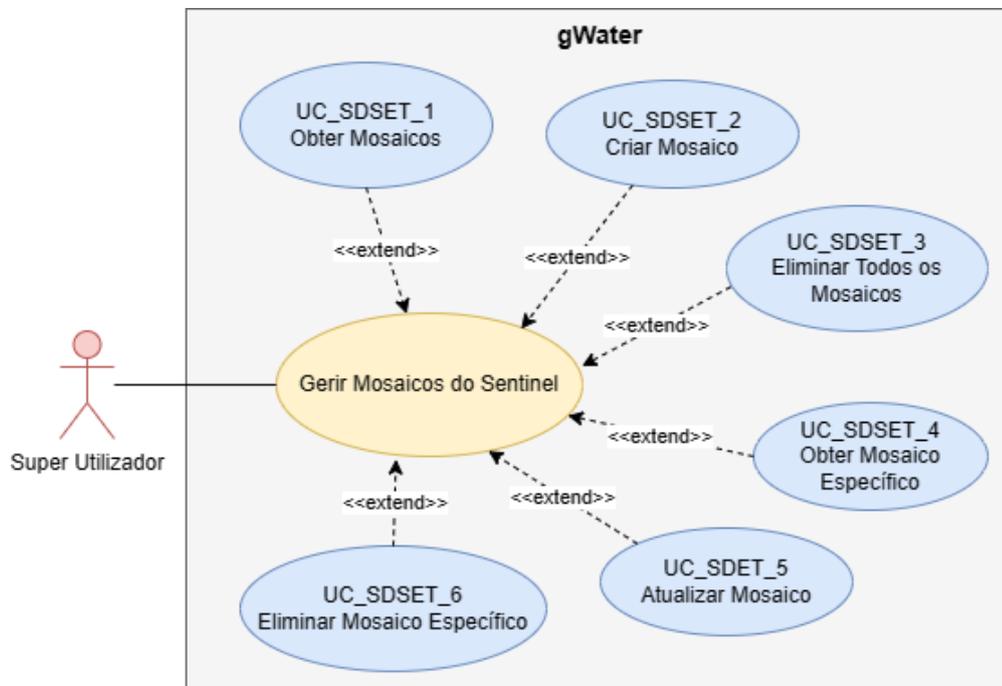


Figura 67 - Diagrama de casos de uso relacionado com a gestão de mosaicos do Sentinel-2.

Tabela 77 - Caso de uso "Obter Mosaicos do Sentinel-2".

UC_SDSET_1 Obter Mosaicos Sentinel-2	
Requisitos Associados	RF047
Endpoints Associados	E_SDSET_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todos os mosaicos do Sentinel-2 existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão dos mosaicos Sentinel-2. 3) Visualiza a lista completa de mosaicos Sentinel-2 existentes no sistema.
Cenários alternativo	Não existe.

Tabela 78 - Caso de uso "Criar Mosaico do Sentinel-2".

UC_SDSET_2 Criar Mosaico Sentinel-2	
Requisitos Associados	RF047
Endpoints Associados	E_SDSET_2
Ator	Superuser
Objetivo	O utilizador pretende criar um mosaico Sentinel-2 no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos mosaicos Sentinel-2. 3) Clica em adicionar mosaico. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria um mosaico Sentinel-2. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 79 - Caso de uso "Eliminar Mosaicos do Sentinel-2".

UC_SDSET_3 Eliminar Mosaicos Sentinel-2	
Requisitos Associados	RF047
Endpoints Associados	E_SDSET_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todos os mosaicos Sentinel-2 existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos mosaicos Sentinel-2. 3) Clica na opção para eliminar todos os mosaicos Sentinel-2. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina os mosaicos Sentinel-2 e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 80 - Caso de uso "Obter Mosaico do Sentinel-2 Específico".

UC_SDSET_4 Obter Mosaico Sentinel-2 Específico	
Requisitos Associados	RF047
Endpoints Associados	E_SDSET_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar um mosaico Sentinel-2 específico existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos mosaicos Sentinel-2. 3) Escolhe um mosaico Sentinel-2 na lista completa de mosaicos do Sentinel-2. 4) O sistema devolve todas as informações relacionadas com esse mosaico do Sentinel-2.
Cenários alternativo	Não existe.

Tabela 81 - Caso de uso "Editar Mosaico do Sentinel-2".

UC_SDSET_5 Editar Mosaico Sentinel-2	
Requisitos Associados	RF047
Endpoints Associados	E_SDSET_5
Ator	Superuser
Objetivo	O utilizador pretende editar um mosaico Sentinel-2 no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos mosaicos Sentinel-2. 3) Clica num mosaico específico na lista de mosaicos Sentinel-2 existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza o mosaico Sentinel-2. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 82 - Caso de uso "Eliminar Mosaico do Sentinel-2 Especifico".

UC_SDSET_6 Eliminar Mosaico Sentinel-2 Especifico	
Requisitos Associados	RF047
Endpoints Associados	E_SDSET_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar um mosaico Sentinel-2 específico existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos mosaicos do Sentinel-2. 3) Escolhe um mosaico Sentinel-2 na lista completa de mosaicos do Sentinel-2. 4) O sistema devolve todas as informações relacionadas com esse mosaico do Sentinel-2. 5) Clica na opção para eliminar o mosaico do Sentinel-2. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina o mosaico Sentinel-2 e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão dos *Algae Bloom Datasets*

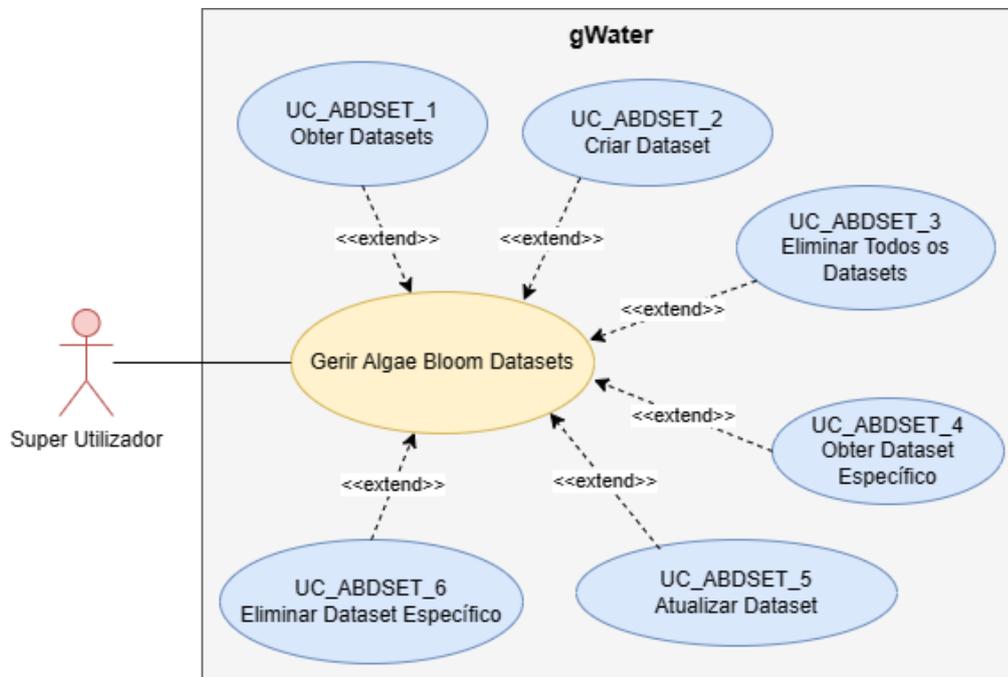


Figura 68 - Diagrama de casos de uso relacionado com a gestão dos *Algae Bloom Datasets*.

Tabela 83 - Caso de uso "Obter Datasets *Algae Bloom*".

UC_ABDSET_1 Obter Datasets <i>Algae Bloom</i>	
Requisitos Associados	RF049
Endpoints Associados	E_ABDSET_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todos os <i>Algae Bloom Datasets</i> existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão dos Datasets <i>Algae Bloom</i> . 3) Visualiza a lista completa de Datasets <i>Algae Bloom</i> existentes no sistema.
Cenários alternativo	Não existe.

Tabela 84 - Caso de uso "Criar um *Algae Bloom* Dataset".

UC_ABDSET_2 Criar um <i>Algae Bloom</i> Dataset	
Requisitos Associados	RF049
Endpoints Associados	E_ABDSET_2
Ator	Superuser
Objetivo	O utilizador pretende criar um <i>Algae Bloom</i> Dataset no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos <i>Algae Bloom</i> Datasets. 3) Clica em adicionar Dataset. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria um <i>Algae Bloom</i> Dataset. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 85 - Caso de uso "Eliminar *Algae Bloom* Datasets".

UC_ABDSET_3 Eliminar <i>Algae Bloom</i> Datasets	
Requisitos Associados	RF049
Endpoints Associados	E_ABDSET_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todos os <i>Algae Bloom</i> Datasets existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos <i>Algae Bloom</i> Datasets. 3) Clica na opção para eliminar todos os <i>Algae Bloom</i> Datasets. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina os <i>Algae Bloom</i> Datasets e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 86 - Caso de uso "Obter um *Algae Bloom* Dataset Específico".

UC_ABDSET_4 Obter <i>Algae Bloom</i> Dataset Específico	
Requisitos Associados	RF049
Endpoints Associados	E_ABDSET_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar um <i>Algae Bloom</i> Dataset específico existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos <i>Algae Bloom</i> Datasets. 3) Escolha um <i>Algae Bloom</i> Dataset na lista completa de Datasets <i>Algae Bloom</i>. 4) O sistema devolve todas as informações relacionadas com esse <i>Algae Bloom</i> Dataset.
Cenários alternativo	Não existe.

Tabela 87 - Caso de uso "Editar um *Algae Bloom* Dataset".

UC_ABDSET_5 Editar <i>Algae Bloom</i> Dataset	
Requisitos Associados	RF049
Endpoints Associados	E_ABSET_5
Ator	Superuser
Objetivo	O utilizador pretende editar um <i>Algae Bloom</i> Dataset no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão dos <i>Algae Bloom</i> Datasets. 3) Clica num Dataset específico na lista de <i>Algae Bloom</i> Datasets existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza o <i>Algae Bloom</i> Dataset. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 88 - Caso de uso "Eliminar um *Algae Bloom* Dataset Específico".

UC_ABDSET_6 Eliminar <i>Algae Bloom</i> Dataset Específico	
Requisitos Associados	RF049
Endpoints Associados	E_ABSET_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar um <i>Algae Bloom</i> Dataset específico existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão <i>Algae Bloom</i> Datasets. 3) Escolhe um <i>Algae Bloom</i> Dataset na lista completa de <i>Algae Bloom</i> Datasets. 4) O sistema devolve todas as informações relacionadas com esse <i>Algae Bloom</i> Dataset. 5) Clica na opção para eliminar o <i>Algae Bloom</i> Dataset. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina o <i>Algae Bloom</i> Dataset e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Camadas *Algae Bloom*

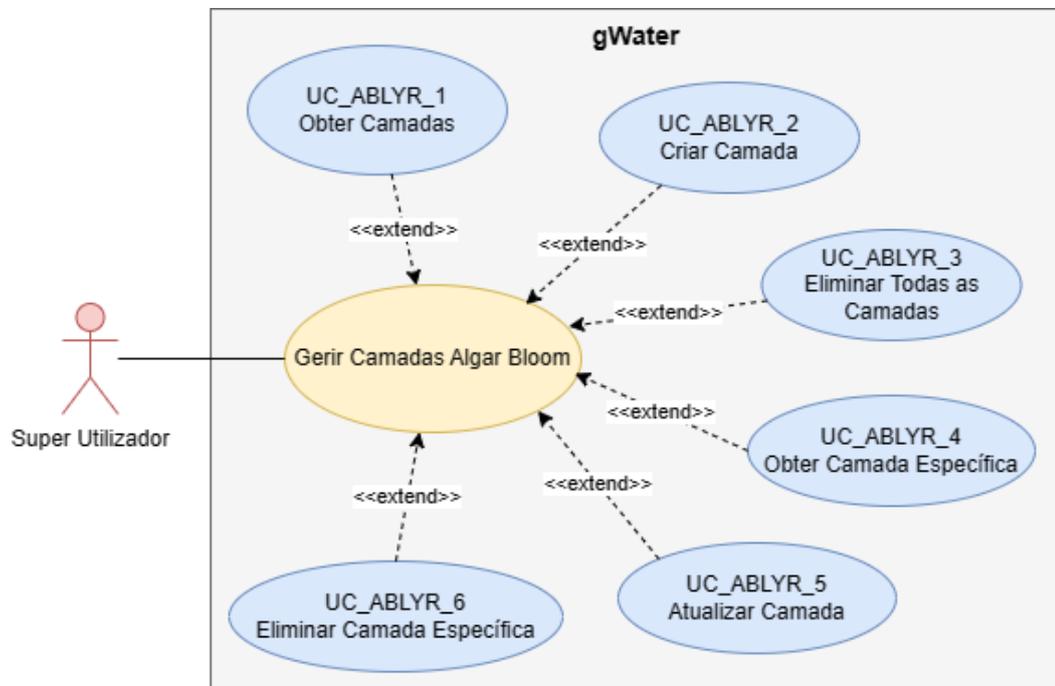


Figura 69 - Diagrama de casos de uso relacionado com a gestão de camadas *Algae Bloom*.

Tabela 89 - Caso de uso "Obter Camadas *Algae Bloom*".

UC_ABLR_1 Obter Camadas <i>Algae Bloom</i>	
Requisitos Associados	RF050
Endpoints Associados	E_ABLR_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as camadas <i>Algae Bloom</i> existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das camadas <i>Algae Bloom</i>. 3) Visualiza a lista completa de camadas <i>Algae Bloom</i> existentes no sistema.
Cenários alternativo	Não existe.

Tabela 90 - Caso de uso "Criar Camada *Algae Bloom*".

UC_ABLYR_2 Criar Camada <i>Algae Bloom</i>	
Requisitos Associados	RF050
Endpoints Associados	E_ABLYR_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma camada <i>Algae Bloom</i> no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas <i>Algae Bloom</i>. 3) Clica em adicionar camada. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma camada <i>Algae Bloom</i>. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> b) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 91 - Caso de uso "Eliminar Camadas *Algae Bloom*".

UC_ABLYR_3 Eliminar Camadas <i>Algae Bloom</i>	
Requisitos Associados	RF050
Endpoints Associados	E_ABLYR_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as camadas <i>Algae Bloom</i> existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas <i>Algae Bloom</i>. 3) Clica na opção para eliminar todas as camadas <i>Algae Bloom</i>. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as camadas <i>Algae Bloom</i> e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 92 - Caso de uso "Obter Camada *Algae Bloom* Específica".

UC_ABLYR_4 Obter Camada <i>Algae Bloom</i> Específica	
Requisitos Associados	RF050
Endpoints Associados	E_ABLYR_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma camada <i>Algae Bloom</i> específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas <i>Algae Bloom</i>. 3) Escolha uma camada <i>Algae Bloom</i> na lista completa de camadas <i>Algae Bloom</i>. 4) O sistema devolve todas as informações relacionadas com essa camada <i>Algae Bloom</i>.
Cenários alternativo	Não existe.

Tabela 93 - Caso de uso "Editar Camada *Algae Bloom*".

UC_ABLYR_5 Editar Camada <i>Algae Bloom</i>	
Requisitos Associados	RF050
Endpoints Associados	E_ABLYR_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma camada <i>Algae Bloom</i> no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas <i>Algae Bloom</i>. 3) Clica numa camada específica na lista de camadas <i>Algae Bloom</i> existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza a camada <i>Algae Bloom</i>. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 94 - Caso de uso "Eliminar Camada *Algae Bloom* Específica".

UC_ABLYR_6 Eliminar Camada <i>Algae Bloom</i> Específica	
Requisitos Associados	RF050
Endpoints Associados	E_ABLYR_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma camada <i>Algae Bloom</i> específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das camadas <i>Algae Bloom</i>. 3) Escolhe uma camada <i>Algae Bloom</i> na lista completa de camadas <i>Algae Bloom</i>. 4) O sistema devolve todas as informações relacionadas com essa camada <i>Algae Bloom</i>. 5) Clica na opção para eliminar a camada <i>Algae Bloom</i>. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a camada <i>Algae Bloom</i> e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Camadas CMEMS

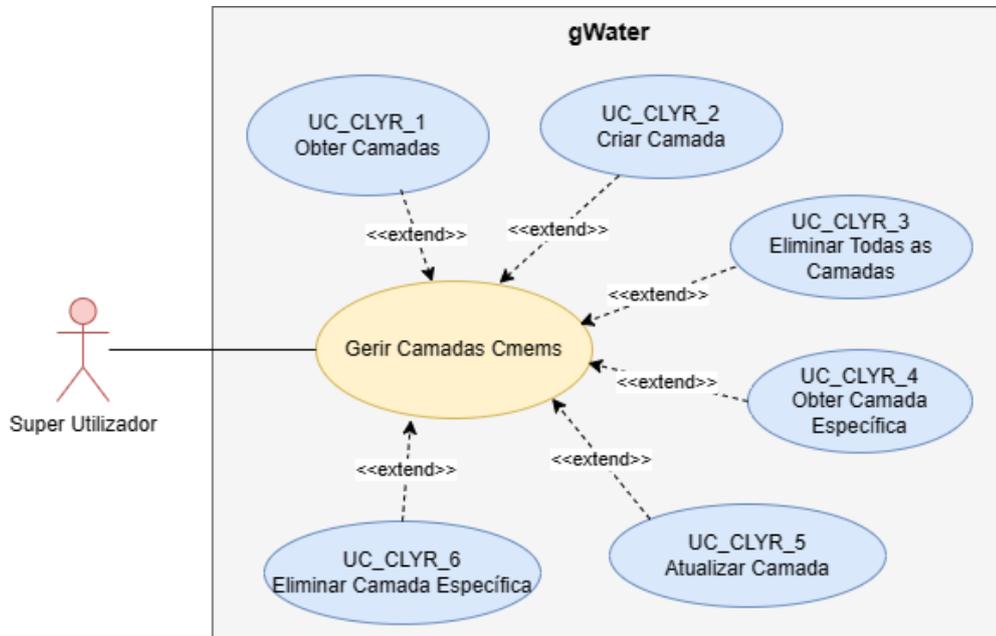


Figura 70 - Diagrama de casos de uso relacionado com a gestão das camadas CMEMS do sistema gWater.

Tabela 95 - Caso de uso "Obter Camadas CMEMS".

UC_CLYR_1 Obter Camadas CMEMS	
Requisitos Associados	RF048
Endpoints Associados	E_CLYR_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as camadas CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas CMEMS. 3) Visualiza a lista completa de camadas CMEMS existentes no sistema.
Cenários alternativo	Não existe.

Tabela 96 - Caso de uso "Criar Camada CMEMS".

UC_CLYR_2 Criar Camada CMEMS	
Requisitos Associados	RF048
Endpoints Associados	E_CLYR_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma camada CMEMS no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas CMEMS. 3) Clica em adicionar camada. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma camada CMEMS. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 97 - Caso de uso "Eliminar Camadas CMEMS".

UC_CLYR_3 Eliminar Camadas CMEMS	
Requisitos Associados	RF048
Endpoints Associados	E_CLYR_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as camadas CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas CMEMS. 3) Clica na opção para eliminar todas as camadas CMEMS. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as camadas CMEMS e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 98 - Caso de uso "Obter Camada CMEMS Específica".

UC_CLYR_4 Obter Camada CMEMS Específica	
Requisitos Associados	RF048
Endpoints Associados	E_CLYR_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma camada CMEMS específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas CMEMS. 3) Escolha uma camada CMEMS na lista completa de camadas CMEMS. 4) O sistema devolve todas as informações relacionadas com essa camada CMEMS.
Cenários alternativo	Não existe.

Tabela 99 - Caso de uso "Editar Camada CMEMS".

UC_CLYR_5 Editar Camada CMEMS	
Requisitos Associados	RF048
Endpoints Associados	E_CLYR_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma camada CMEMS do sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas CMEMS. 3) Clica numa camada específica na lista de camadas CMEMS existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza a camada CMEMS. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 100 - Caso de uso "Eliminar Camada CMEMS Especifica".

UC_CLYR_6 Eliminar Camada CMEMS Especifica	
Requisitos Associados	RF048
Endpoints Associados	E_CLYR_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma camada CMEMS específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas CMEMS. 3) Escolhe uma camada CMEMS na lista completa de camadas CMEMS. 4) O sistema devolve todas as informações relacionadas com essa camada CMEMS. 5) Clica na opção para eliminar a camada CMEMS. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a camada CMEMS e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Camadas Sentinel-2

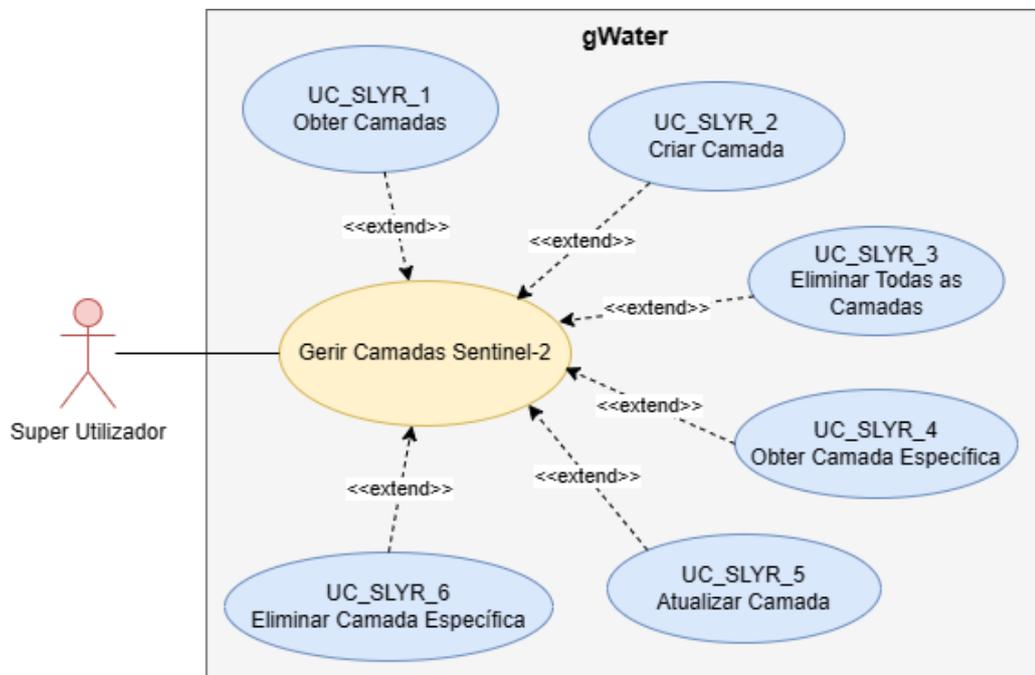


Figura 71 - Diagrama de casos de uso relacionado com a gestão das camadas Sentinel-2 do sistema gWater.

Tabela 101 - Caso de uso "Obter Camadas Sentinel-2".

UC_SLYR_1 Obter Camadas Sentinel-2	
Requisitos Associados	RF051
Endpoints Associados	E_SLYR_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as camadas Sentinel-2 existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das camadas Sentinel-2. 3) Visualiza a lista completa de camadas Sentinel-2 existentes no sistema.
Cenários alternativo	Não existe.

Tabela 102 - Caso de uso "Criar Camada Sentinel-2".

UC_SLYR_2 Criar Camada Sentinel-2	
Requisitos Associados	RF051
Endpoints Associados	E_SLYR_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma camada Sentinel-2 no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das camadas Sentinel-2. 3) Clica em adicionar camada. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma camada Sentinel-2. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> c) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 103 - Caso de uso "Eliminar Camadas Sentinel-2".

UC_SLYR_3 Eliminar Camadas Sentinel-2	
Requisitos Associados	RF051
Endpoints Associados	E_SLYR_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as camadas Sentinel-2 existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das camadas Sentinel-2. 3) Clica na opção para eliminar todas as camadas Sentinel-2. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as camadas Sentinel-2 e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 104 - Caso de uso "Obter Camada Sentinel-2 Específica".

UC_SLYR_4 Obter Camada Sentinel-2 Específica	
Requisitos Associados	RF051
Endpoints Associados	E_SLYR_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma camada Sentinel-2 específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas Sentinel-2. 3) Escolha uma camada Sentinel-2 na lista completa de camadas Sentinel-2. 4) O sistema devolve todas as informações relacionadas com essa camada Sentinel-2.
Cenários alternativo	Não existe.

Tabela 105 - Caso de uso "Editar Camada Sentinel-2".

UC_SLYR_5 Editar Camada Sentinel-2	
Requisitos Associados	RF051
Endpoints Associados	E_SLYR_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma camada Sentinel-2 no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas Sentinel-2. 3) Clica numa camada específica na lista de camadas Sentinel-2 existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza a camada Sentinel-2. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 106 - Caso de uso "Eliminar Camada Sentinel-2 Específica".

UC_SLYR_6 Eliminar Camada Sentinel-2 Específica	
Requisitos Associados	RF051
Endpoints Associados	E_SLYR_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma camada Sentinel-2 específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das camadas Sentinel-2. 3) Escolhe uma camada Sentinel-2 na lista completa de camadas Sentinel-2. 4) O sistema devolve todas as informações relacionadas com essa camada Sentinel-2. 5) Clica na opção para eliminar a camada Sentinel-2. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a camada Sentinel-2 e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Legendas dos Dados CMEMS

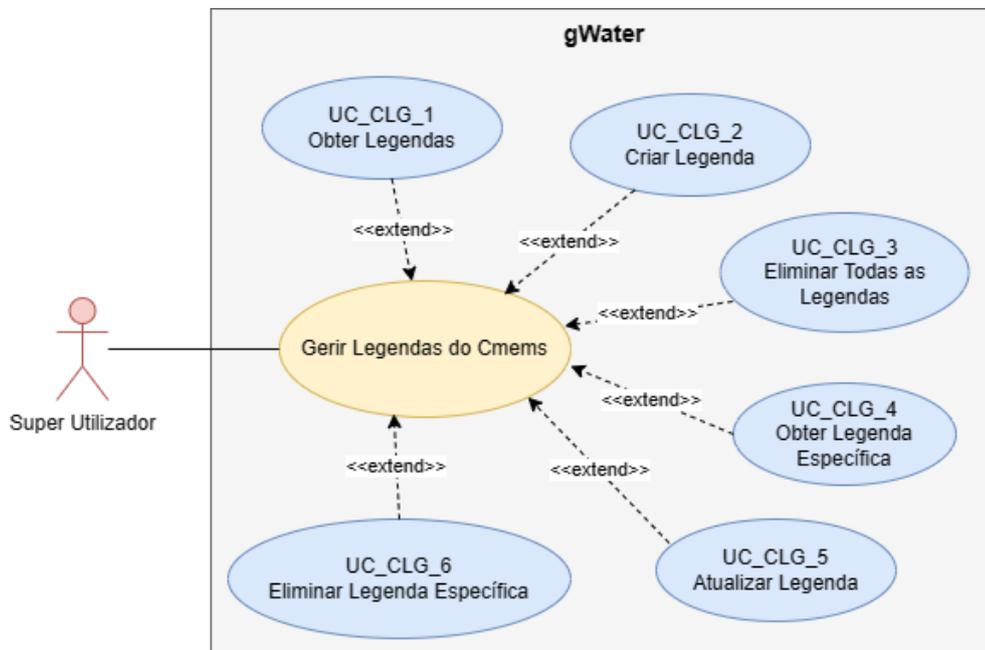


Figura 72 - Diagrama de casos de uso relacionado com a gestão das legendas das camadas CMEMS.

Tabela 107 - Caso de uso "Obter Legendas CMEMS".

UC_CLG_1 Obter Legendas CMEMS	
Requisitos Associados	RF052
Endpoints Associados	E_CLG_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as legendas CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das legendas CMEMS. 3) Visualiza a lista completa de legendas CMEMS existentes no sistema.
Cenários alternativo	Não existe.

Tabela 108 - Caso de uso "Criar Legenda CMEMS".

UC_CLG_2 Criar Legenda CMEMS	
Requisitos Associados	RF052
Endpoints Associados	E_CLG_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma legenda CMEMS no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das legendas CMEMS. 3) Clica em adicionar legenda. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma legenda CMEMS. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> d) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 109 - Caso de uso "Eliminar Legendas CMEMS".

UC_CLG_3 Eliminar Legendas CMEMS	
Requisitos Associados	RF052
Endpoints Associados	E_CLG_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as legendas CMEMS existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das legendas CMEMS. 3) Clica na opção para eliminar todas as legendas CMEMS. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as legendas CMEMS e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 110 - Caso de uso "Obter Legenda CMEMS Específica".

UC_CLG_4 Obter Legenda CMEMS Específica	
Requisitos Associados	RF052
Endpoints Associados	E_CLG_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma legenda CMEMS específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das legendas CMEMS. 3) Escolha uma legenda CMEMS na lista completa de legendas CMEMS. 4) O sistema devolve todas as informações relacionadas com essa legenda CMEMS.
Cenários alternativo	Não existe.

Tabela 111 - Caso de uso "Editar Legenda CMEMS".

UC_CLG_5 Editar Legenda CMEMS	
Requisitos Associados	RF052
Endpoints Associados	E_CLG_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma legenda CMEMS no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das legendas CMEMS. 3) Clica numa legenda específica na lista de legendas CMEMS existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza a legenda CMEMS. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 112 - Caso de uso "Eliminar Legenda CMEMS Específica".

UC_CLG_6 Eliminar Legenda CMEMS Específica	
Requisitos Associados	RF052
Endpoints Associados	E_CLG_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma legenda CMEMS específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das legendas CMEMS. 3) Escolhe uma legenda CMEMS na lista completa de legendas CMEMS. 4) O sistema devolve todas as informações relacionadas com essa legenda CMEMS. 5) Clica na opção para eliminar a legenda CMEMS. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a legenda CMEMS e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Gestão das Áreas de Interesse

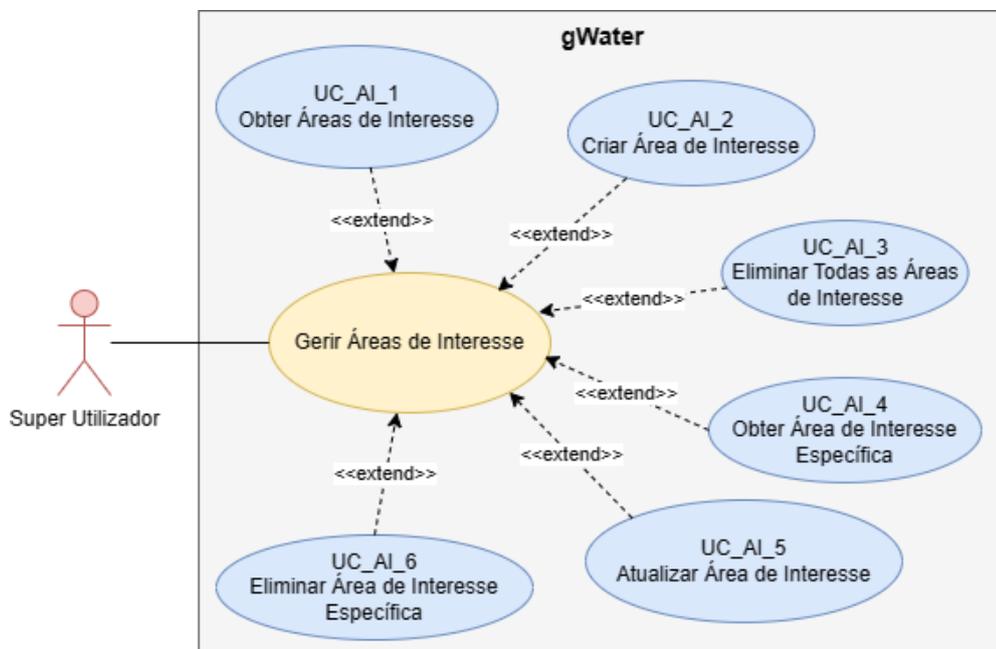


Figura 73 - Diagrama de casos de uso relacionado com a gestão das áreas de interesse do sistema gWater.

Tabela 113 - Caso de uso "Obter Áreas de Interesse".

UC_AI_1 Obter Áreas de Interesse	
Requisitos Associados	RF042
Endpoints Associados	E_AI_1
Ator	Superuser
Objetivo	O utilizador pretende visualizar todas as áreas de interesse existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	1) O utilizador acede à página de gestão. 2) Accede à secção relacionada com a gestão das áreas de interesse. 3) Visualiza a lista completa de áreas de interesse existentes no sistema.
Cenários alternativo	Não existe.

Tabela 114 - Caso de uso "Criar Área de Interesse".

UC_AI_2 Criar Área de Interesse	
Requisitos Associados	RF042
Endpoints Associados	E_AI_2
Ator	Superuser
Objetivo	O utilizador pretende criar uma área de interesse no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das áreas de interesse. 3) Clica no botão para adicionar área de interesse. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e cria uma área de interesse. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> e) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 115 - Caso de uso "Eliminar Áreas de Interesse".

UC_AI_3 Eliminar Áreas de Interesse	
Requisitos Associados	RF042
Endpoints Associados	E_AI_3
Ator	Superuser
Objetivo	O utilizador pretende eliminar todas as áreas de interesse existentes no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das áreas de interesse. 3) Clica na opção para eliminar todas as áreas de interesse. 4) O sistema pergunta se o utilizador tem a certeza. 5) O utilizador responde que sim. 6) O sistema elimina as áreas de interesse e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Tabela 116 - Caso de uso "Obter Área de Interesse Específica".

UC_AI_4 Obter Área de Interesse Específica	
Requisitos Associados	RF042
Endpoints Associados	E_AI_4
Ator	Superuser
Objetivo	O utilizador pretende visualizar uma área de interesse específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das áreas de interesse. 3) Escolha uma área de interesse na lista completa de áreas de interesse. 4) O sistema devolve todas as informações relacionadas com essa área de interesse.
Cenários alternativo	Não existe.

Tabela 117 - Caso de uso "Editar Área de Interesse".

UC_AI_5 Editar Área de Interesse	
Requisitos Associados	RF042
Endpoints Associados	E_AI_5
Ator	Superuser
Objetivo	O utilizador pretende editar uma área de interesse no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das áreas de interesse. 3) Clica numa área de interesse específica na lista de áreas de interesse existentes no sistema. 4) Fornece todos os dados necessários e submete o formulário. 5) O sistema valida os dados introduzidos e atualiza a área de interesse. 6) O sistema informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 5) O utilizador insere dados inválidos. <ol style="list-style-type: none"> a) O sistema apresenta uma mensagem com informação acerca do erro.

Tabela 118 - Caso de uso "Eliminar Área de Interesse Específica".

UC_AI_6 Eliminar Área de Interesse Específica	
Requisitos Associados	RF042
Endpoints Associados	E_AI_6
Ator	Superuser
Objetivo	O utilizador pretende eliminar uma área de interesse específica existente no sistema.
Pré-condições	O utilizador tem de estar autenticado e possuir as permissões necessárias.
Cenário principal	<ol style="list-style-type: none"> 1) O utilizador acede à página de gestão. 2) Acede à secção relacionada com a gestão das áreas de interesse. 3) Escolhe uma área de interesse na lista completa de áreas de interesse. 4) O sistema devolve todas as informações relacionadas com essa área de interesse. 5) Clica na opção para eliminar a área de interesse. 6) O sistema pergunta se o utilizador tem a certeza. 7) O utilizador responde que sim. 8) O sistema elimina a área de interesse e informa o utilizador acerca do sucesso da operação.
Cenários alternativo	<ol style="list-style-type: none"> 7) O utilizador responde que não. <ol style="list-style-type: none"> a) O sistema não efetua a operação.

Apêndice B: Endpoints

Tabela 119 - *Endpoints* relacionados com a autenticação no sistema gWater.

<i>Endpoints</i> Autenticação			
ID	URL	Método	Descrição
E_A_1	/authapi/token/<str:op>/	POST	Cria um <i>token</i> de acesso.
E_A_2	/authapi/change-password/	PUT	Altera a palavra-passe de um utilizador que faça o pedido.
E_A_3	/authapi/password-recovery/	PUT	Envia um email para o utilizador recuperar a sua palavra-passe.

Tabela 120 - *Endpoints* relacionados com a gestão dos utilizadores.

<i>Endpoints</i> Gestão de Utilizadores			
ID	URL	Método	Descrição
E_GU_1	/authapi/users/	GET	Devolve todos os utilizadores.
E_GU_2	/authapi/users/	POST	Adiciona um utilizador.
E_GU_3	/authapi/users/	DELETE	Elimina todos os utilizadores.
E_GU_4	/authapi/user/<str:userid>/	GET	Devolve um utilizador específico.
E_GU_5	/authapi/user/<str:userid>/	PUT	Atualiza um utilizador.
E_GU_6	/authapi/user/<str:userid>/	DELETE	Elimina um utilizador.

Tabela 121 - *Endpoints* relacionados com os pedidos ao GeoServer.

<i>Endpoints</i> MapLayer			
ID	URL	Método	Descrição
E_MP_1	/geosrv/WMS/<str:work>/	GET	Devolve um WMS.
E_MP_2	/geosrv/WFS/<str:work>/<str:lyr>/	GET	Devolve um WFS.

Tabela 122 - *Endpoints* relacionados com os gráficos.

<i>Endpoints</i> Gráficos			
ID	URL	Método	Descrição
E_G_1	/charts/histogram/	GET	Devolve os dados necessários para a construção do histograma.
E_G_2	/charts/scatterplot/	POST	Devolve os dados necessários para a construção do gráfico de dispersão.
E_G_3	/charts/tempevolution/	POST	Devolve os dados necessários para a construção do gráfico de evolução temporal.

Tabela 123 - *Endpoints* relacionados com as ferramentas.

<i>Endpoints</i> Ferramentas			
ID	URL	Método	Descrição
E_F_1	/tools/tools/	GET	Devolve todas as ferramentas existentes no sistema.
E_F_2	/tools/tools/	POST	Adiciona uma ferramenta ao sistema.
E_F_3	/tools/tools/	DELETE	Elimina todas as ferramentas.
E_F_4	/tools/tools/<str:alias>/	GET	Devolve uma ferramenta específica.
E_F_5	/tools/tools/<str:alias>/	PUT	Atualiza uma ferramenta.
E_F_6	/tools/tools/<str:alias>/	DELETE	Elimina uma ferramenta.

Tabela 124 - *Endpoints* relacionados com os parâmetros das ferramentas.

<i>Endpoints</i> Parâmetros Ferramentas			
ID	URL	Método	Descrição
E_PF_1	/tools/params/	GET	Devolve todos os parâmetros de ferramentas existentes no sistema.
E_PF_2	/tools/params/	POST	Adiciona um parâmetro associado a determinada ferramenta do sistema.
E_PF_3	/tools/params/	DELETE	Elimina todos os parâmetros.
E_PF_4	/tools/param/<str:alias>/	GET	Devolve um parâmetro específico.
E_PF_5	/tools/param/<str:alias>/	PUT	Atualiza um parâmetro.
E_PF_6	/tools/param/<str:alias>/	DELETE	Elimina um parâmetro.

Tabela 125 - *Endpoints* relacionados com a execução de ferramentas.

<i>Endpoints</i> “Running Tools”			
ID	URL	Método	Descrição
E_RT_1	/tools/runtools/	GET	Devolve todas as ferramentas executadas no sistema.
E_RT_2	/tools/runtools/	POST	Adiciona uma execução de ferramenta ao sistema.
E_RT_3	/tools/runtools/	DELETE	Elimina todos os registos de execuções de ferramentas.
E_RT_4	/tools/runtool/<str:alias>/	GET	Devolve uma execução de ferramenta específica.
E_RT_5	/tools/runtool/<str:alias>/	PUT	Atualiza um registo de execução de ferramenta.
E_RT_6	/tools/runtool/<str:alias>/	DELETE	Elimina um registo de execução de ferramenta.

Tabela 126 - *Endpoints* relacionados o resultado da execução de ferramentas.

Endpoints “Running Results”			
ID	URL	Método	Descrição
E_RR_1	/tools/results/	GET	Devolve todos os resultados de execuções de ferramentas.
E_RR_2	/tools/results/	POST	Adiciona um resultado de execução de ferramenta.
E_RR_3	/tools/results/	DELETE	Elimina todos os registos de resultados de execuções de ferramentas.
E_RR_4	/tools/result/<int_fid>/	GET	Devolve um resultado de execução de ferramenta específica.
E_RR_5	/tools/result/<int:fid>/	PUT	Atualiza um registo de resultado de execução de ferramenta.
E_RR_6	/tools/result/<int:fid>/	DELETE	Elimina um registo de resultado de execução de ferramenta.

Tabela 127 - Outros *endpoints* relacionados com a execução de ferramentas.

Outros <i>endpoints</i> relacionados com execução de ferramentas			
ID	URL	Método	Descrição
E_OEF_1	/tools/runtools/runtool-vis-raster/<int:rtoolid>/	PUT	Guarda o resultado de uma execução de ferramentas (ficheiro).
E_OEF_2	/tools/results-geoms/	GET	Devolve todas as geometrias dos resultados de execuções de ferramentas.
E_OEF_3	/tools/results-geoms/	POST	Adiciona uma geometria de um resultado de execução de ferramenta.
E_OEF_4	/tools/results-geoms/	DELETE	Elimina todos os registos de geometrias de resultados de execuções de ferramentas.

Tabela 128 - *Endpoints* relacionados com os estilos das camadas CMEMS.

<i>Endpoints</i> Estilos CMEMS do Sistema			
ID	URL	Método	Descrição
E_ES_1	/geovis/cmems-styles/	GET	Devolve todos os estilos existentes no sistema.
E_ES_2	/geovis/cmems-styles/	POST	Adiciona um estilo ao sistema.
E_ES_3	/geovis/cmems-styles/	DELETE	Elimina todos os estilos do sistema.
E_ES_4	/geovis/cmems-style/<str:slug>/	GET	Devolve um estilo específico do sistema.
E_ES_5	/geovis/cmems-style/<str:slug>/	PUT	Atualiza um estilo.
E_ES_6	/geovis/cmems-style/<str:slug>/	DELETE	Elimina um estilo.

Tabela 129 - *Endpoints* relacionados com os estilos do GeoServer.

<i>Endpoints</i> Estilos GeoServer			
ID	URL	Método	Descrição
E_EG_1	/geosrv/styles/	GET	Devolve todos os estilos existentes no GeoServer.
E_EG_2	/geosrv/styles/	POST	Adiciona um estilo ao GeoServer.
E_EG_3	/geosrv/style/<str:style>/	PUT	Atualiza um estilo do GeoServer.
E_EG_4	/geosrv/style/<str:style>/	DELETE	Elimina um estilo do GeoServer.

Tabela 130 - *Endpoints* relacionados com os workspaces do GeoServer.

Endpoints Workspaces GeoServer			
ID	URL	Método	Descrição
E_WS_1	/geosrv/workspaces/	GET	Devolve todos os <i>workspaces</i> existentes no GeoServer.
E_WS_2	/geosrv/workspaces/	POST	Adiciona um <i>workspace</i> ao GeoServer.
E_WS_3	/geosrv/workspace/<str:ws>/	GET	Devolve um <i>workspace</i> específico do GeoServer.
E_WS_4	/geosrv/workspace/<str:ws>/	DELETE	Elimina um <i>workspace</i> do GeoServer.

Tabela 131 - *Endpoints* relacionados com as stores do GeoServer.

Endpoints Stores GeoServer			
ID	URL	Método	Descrição
E_ST_1	/geosrv/<str:ws>/stores/	GET	Devolve todas as <i>stores</i> existentes no GeoServer.
E_ST_2	/geosrv/<str:ws>/stores/	POST	Adiciona uma <i>store</i> ao GeoServer.
E_ST_3	/geosrv/store/<str:st>/	GET	Devolve uma <i>store</i> específica do GeoServer.
E_ST_4	/geosrv/store/<str:st>/	DELETE	Elimina uma <i>store</i> do GeoServer.

Tabela 132 - *Endpoints* relacionados com as camadas do GeoServer.

Endpoints Camadas GeoServer			
ID	URL	Método	Descrição
E_GLYR_1	/geosrv/layers/	GET	Devolve todas as camadas existentes no GeoServer.
E_GLYR_2	/geosrv/layers/	POST	Adiciona uma camada ao GeoServer.
E_GLYR_3	/geosrv/layer/<str:lyr>/	GET	Devolve uma camada específica do GeoServer.
E_GLYR_4	/geosrv/layer/<str:lyr>/	PUT	Atualiza uma camada do GeoServer.
E_GLYR_5	/geosrv/layer/<str:lyr>/	DELETE	Elimina uma camada do GeoServer.

Tabela 133 - *Endpoints* relacionados com as variáveis.

Endpoints Variáveis			
ID	URL	Método	Descrição
E_VAR_1	/dset/variables/	GET	Devolve todas as variáveis existentes no sistema.
E_VAR_2	/dset/variables/	POST	Adiciona uma variável ao sistema.
E_VAR_3	/dset/variables/	DELETE	Elimina todas as variáveis do sistema.
E_VAR_4	/dset/ variable/<str:varslug>/	GET	Devolve uma variável específica do sistema.
E_VAR_5	/dset/ variable/<str:varslug>/	PUT	Atualiza uma variável.
E_VAR_6	/dset/ variable/<str:varslug>/	DELETE	Elimina uma variável.

Tabela 134 - *Endpoints* relacionados com as variáveis do CMEMS.

Endpoints Variáveis CMEMS			
ID	URL	Método	Descrição
E_VC_1	/dset/cmems-variables/	GET	Devolve todas as variáveis CMEMS existentes no sistema.
E_VC_2	/dset/cmems-variables/	POST	Adiciona uma variável CMEMS ao sistema.
E_VC_3	/dset/cmems-variables/	DELETE	Elimina todas as variáveis CMEMS do sistema.
E_VC_4	/dset/cmems- variable/<str:slug>/	GET	Devolve uma variável CMEMS específica do sistema.
E_VC_5	/dset/cmems- variable/<str:slug>/	PUT	Atualiza uma variável CMEMS.
E_VC_6	/dset/cmems- variable/<str:slug>/	DELETE	Elimina uma variável CMEMS.

Tabela 135 - *Endpoints* relacionados com os mosaicos do Sentinel-2.

Endpoints Mosaicos do Sentinel-2			
ID	URL	Método	Descrição
E_SDSET_1	/dset/Sentinel-2-dsets/	GET	Devolve todos os mosaicos Sentinel-2 existentes no sistema.
E_SDSET_2	/dset/Sentinel-2-dsets/	POST	Adiciona um mosaico Sentinel-2 ao sistema.
E_SDSET_3	/dset/Sentinel-2-dsets/	DELETE	Elimina todos os mosaicos Sentinel-2 do sistema.
E_SDSET_4	/dset/Sentinel-2-dset/<int:fid>/	GET	Devolve um mosaico Sentinel-2 específico do sistema.
E_SDSET_5	/dset/Sentinel-2-dset/<int:fid>/	PUT	Atualiza um mosaico do Sentinel-2.
E_SDSET_6	/dset/Sentinel-2-dset/<int:fid>/	DELETE	Elimina um mosaico do Sentinel-2.

Tabela 136 - *Endpoints* relacionados com os *Algae Bloom Datasets*.

Endpoints Algae Bloom Datasets			
ID	URL	Método	Descrição
E_ABDSET_1	/dset/ab-datasets/	GET	Devolve todos os <i>Algae Bloom Datasets</i> existentes no sistema.
E_ABDSET_2	/dset/ab-datasets/	POST	Adiciona um <i>Algae Bloom Dataset</i> ao sistema.
E_ABDSET_3	/dset/ab-datasets/	DELETE	Elimina todos os <i>Algae Bloom Datasets</i> do sistema.
E_ABDSET_4	/dset/ab-dataset/<int:fid>/	GET	Devolve um <i>Algae Bloom Dataset</i> específico do sistema.
E_ABDSET_5	/dset/ab-dataset/<int:fid>/	PUT	Atualiza um <i>Algae Bloom Dataset</i> .
E_ABDSET_6	/dset/ab-dataset/<int:fid>/	DELETE	Elimina um <i>Algae Bloom Dataset</i> .

Tabela 137 - *Endpoints* relacionados com as camadas *Algae Bloom*.

Endpoints Camadas <i>Algae Bloom</i>			
ID	URL	Método	Descrição
E_ABLYR_1	/geovis/ab-layers/	GET	Devolve todas as camadas <i>Algae Bloom</i> existentes no sistema.
E_ABLYR_2	/geovis/ab-layers/	POST	Adiciona uma camada <i>Algae Bloom</i> ao sistema.
E_ABLYR_3	/geovis/ab-layers/	DELETE	Elimina todas as camadas <i>Algae Bloom</i> do sistema.
E_ABLYR_4	/geovis/ab-layer/<str:alias>/	GET	Devolve uma camada <i>Algae Bloom</i> específica do sistema.
E_ABLYR_5	/geovis/ab-layer/<str:alias>/	PUT	Atualiza uma camada <i>Algae Bloom</i> .
E_ABLYR_6	/geovis/ab-layer/<str:alias>/	DELETE	Elimina uma camada <i>Algae Bloom</i> .

Tabela 138 - *Endpoints* relacionados com as camadas *CMEMS*.

Endpoints Camadas <i>CMEMS</i>			
ID	URL	Método	Descrição
E_CLYR_1	/dset/cmems-layers/	GET	Devolve todos as camadas <i>CMEMS</i> existentes no sistema.
E_CLYR_2	/dset/cmems-layers/	POST	Adiciona uma camada <i>CMEMS</i> ao sistema.
E_CLYR_3	/dset/cmems-layers/	DELETE	Elimina todas as camadas <i>CMEMS</i> do sistema.
E_CLYR_4	/dset/cmems-layer/<int:fid>/	GET	Devolve uma camada <i>CMEMS</i> específica do sistema.
E_CLYR_5	/dset/cmems-layer/<int:fid>/	PUT	Atualiza uma camada <i>CMEMS</i> .
E_CLYR_6	/dset/cmems-layer/<int:fid>/	DELETE	Elimina uma camada <i>CMEMS</i> .

Tabela 139 - *Endpoints* relacionados com as camadas Sentinel-2.

Endpoints Camadas Sentinel-2			
ID	URL	Método	Descrição
E_SLYR_1	/geovis/Sentinel-2-layers/	GET	Devolve todas as camadas Sentinel-2 existentes no sistema.
E_SLYR_2	/geovis/Sentinel-2-layers/	POST	Adiciona uma camada Sentinel-2 ao sistema.
E_SLYR_3	/geovis/Sentinel-2-layers/	DELETE	Elimina todas as camadas Sentinel-2 do sistema.
E_SLYR_4	/geovis/Sentinel-2-layer/<str:slug>/	GET	Devolve uma camada Sentinel-2 específica do sistema.
E_SLYR_5	/geovis/Sentinel-2-layer/<str:slug>/	PUT	Atualiza uma camada Sentinel-2.
E_SLYR_6	/geovis/Sentinel-2-layer/<str:slug>/	DELETE	Elimina uma camada Sentinel-2.

Tabela 140 - *Endpoints* relacionados com as legendas das camadas CMEMS.

Endpoints Legendas CMEMS			
ID	URL	Método	Descrição
E_CLG_1	/geovis/cmems-legends/	GET	Devolve todas as legendas CMEMS existentes no sistema.
E_CLG_2	/geovis/cmems-legends/	POST	Adiciona uma legenda CMEMS ao sistema.
E_CLG_3	/geovis/cmems-legends/	DELETE	Elimina todas as legendas CMEMS do sistema.
E_CLG_4	/geovis/cmems-legend/<str:alias>/	GET	Devolve uma legenda CMEMS específica do sistema.
E_CLG_5	/geovis/cmems-legend/<str:alias>/	PUT	Atualiza uma legenda CMEMS.
E_CLG_6	/geovis/cmems-legend/<str:alias>/	DELETE	Elimina uma legenda CMEMS.

Tabela 141 - *Endpoints* relacionados com as áreas de interesse.

Endpoints Áreas de Interesse			
ID	URL	Método	Descrição
E_AI_1	/authapi/areai/	GET	Devolve todas as áreas de interesse existentes no sistema.
E_AI_2	/authapi/areai/	POST	Adiciona uma área de interesse ao sistema.
E_AI_3	/authapi/areai/	DELETE	Elimina todas as áreas de interesse do sistema.
E_AI_4	/authapi/areai/<str:slug>/	GET	Devolve uma área de interesse específica do sistema.
E_AI_5	/authapi/areai/<str:slug>/	PUT	Atualiza uma área de interesse.
E_AI_6	/authapi/areai/<str:slug>/	DELETE	Elimina uma área de interesse.

Tabela 142 - Outros *endpoints*

Outros Endpoints			
ID	URL	Método	Descrição
E_OE_1	/dset/cmems-layer-raster/<int:dsetid>/	PUT	Envia ficheiro da camada CMEMS para o servidor.
E_OE_2	/dset/cmems-layer-vis-raster/<int:dsetid>/	PUT	Envia ficheiro da camada CMEMS reclassificada para o servidor.
E_OE_3	/dset/sentinel-mos-band/<int:mos>/<str:layer>/	PUT	Envia ficheiro de mosaico do Sentinel-2 para o servidor.
E_OE_4	/dset/sentinel-bandcmp/<int:mos>/	PUT	Envia ficheiro raster composto do Sentinel-2 para o servidor.