



Department of Informatics Engineering

Internship report

Master in Informatics Engineering - Mobile Computing

Earth Observation

Srinivas Kencahppa Singadi

Leiria, September 2017



Department of Informatics Engineering

Internship report

Master in Informatics Engineering - Mobile Computing

Earth Observation

Srinivas Kencahppa Singadi

Master's internship conducted under the guidance of Dr. Antonio Mnauel de Jesus Pereira, Professor of the School of Technology and Management of the Polytechnic Institute of Leiria.

Leiria, September 2017

Dedication

In memory of my father

Esta página foi intencionalmente deixada em branco

Acknowledgement

Foremost, I would like to express my sincere gratitude to my supervisor Doctor Antonio Mnauel de Jesus Pereira for the continuous support of my Internship, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of Internship and writing of this report. I could not have imagined having a better advisor and mentor for my Internship.

Besides my supervisor, I would also like to thank Pedro Borges, from XSealence for proposing a very interesting project and for the support and for their encouragement, insightful comments, and hard questions during the internship. I also want to thank the people at XSealence and Tecmic for having welcomed me with open arms and for creating a friendly environment.

My sincere thanks also goes to my course coordinator Dr. Carlos Fernando Almeida Grilo, for helping me to get an Internship in such a good company.

A thank-you to my colleagues in the daily work, thank you, Joel, João, Ze Pedro, Jóni and André for giving such a wonderful environment to work.

Last but not the least, I would like to thank my family: my parents Kenchappa Singadi and Shanta Singadi, for giving birth and supporting me spiritually throughout my life.

Esta página foi intencionalmente deixada em branco

Abstract

This is a system project report explaining Earth observation project undertaken during my internship with XSealence S.A. The project was proposed by the company but since it was a large project it was divided into small several units and assigned to various groups. The report has been divided into various chapters which discuss the software development life cycle of Earth observation system. the project used agile development model which allowed the project to be divided into various phases. The first phase is discussed in the first chapter which helped state the motivations behind the project. Buoyed by rising demand for marine weather forecast XSealence S.A company developed a marine weather monitoring web application to monitor and predict weather data based on data downloaded from Copernicus marine monitoring service. So as to have an installable system that can be installed on sea vessels XSealence S.A developed a desktop version of the same. This report discusses the development of earth observation system which is the part of the main desktop application.

The aim of developing earth observation was to help predict the marine weather. The project initialization involved setting project aims and goals which were to guide through the development process. The main project goal was to deliver a desktop application that will help forecast marine weather. The application was linked to Copernicus marine monitoring service where it downloaded data in a .nc format and converted to .txt format.

The system requirement phase involved the collection of functional requirements of the system. With the help of my supervisor, it was established my part of the main system only needed to predict marine weather including water and ice speed, water velocity, sea surface height, temperature among other marine weather options. These capabilities of the system helped in system design and development to ensure the final product meets the set targets. UML diagrams including use case, data flow diagrams helped illustrate the flow of information in the system. These designs together with class diagrams were converted into a working solution during system development phase as planned in the project planning phase of the agile methodology.

The system was tested for the achievement of system goals and objectives as stated in system functional requirements. Functional requirements achievement was tested using test use case. The test case tested every functionality of the system to establish the non-achieved goals and objectives. These defects found were communicated to the developer for correction in the subsequent releases. The system has been handed to the XSealence S.A for integration with another main desktop application. The entire system maintenance and deployment will be done by the company. To close the project, documentation was done to cover all activities of the project.

Esta página foi intencionalmente deixada em branco

Resumo

O presente relatório foi dividido em vários capítulos que discutem o ciclo de vida do desenvolvimento de software de um sistema de observação da Terra. O projeto usou um modelo de desenvolvimento ágil que possibilitou a divisão em várias fases. A primeira fase é discutida no primeiro capítulo que ajudou a indicar as motivações por trás do projeto. Impulsionada pela crescente necessidade de uma previsão meteorológica marítima a empresa XSealence S.A desenvolveu uma aplicação web para monitorização e previsão, do estado tempo marítimo, baseada no serviço de monitorização marítimo Copernicus. De modo a ter um sistema que possa ser instalado em embarcações marítimas, a XSealence S.A desenvolveu uma versão desktop do mesmo. Este relatório discute o desenvolvimento de um sistema de observação da Terra, que faz parte da aplicação de desktop principal.

O objetivo do desenvolvimento da aplicação “Observação da Terra” foi ajudar a prever o clima marítimo. A parte inicial do projeto consistiu na definição dos objetivos e metas do projeto que guiarão o processo de desenvolvimento. O principal objetivo do projeto era o de entregar uma aplicação de desktop que ajudaria a prever o clima marítimo. Esta aplicação está conectada ao serviço de monitorização marítima Copernicus, do qual são descarregados os dados em formato “.nc” e convertidos em formato “.txt”.

A fase de requisitos do sistema envolveu a recolha dos requisitos funcionais do sistema. Com a ajuda do meu supervisor, foi estabelecido que a minha parte do sistema principal só era necessária para prever o clima marítimo, incluindo velocidade da água e do gelo, velocidade da água, altura da superfície do mar, temperatura, entre outras opções, de tempo marítimo. Estas capacidades do sistema ajudaram na conceção e desenvolvimento do mesmo de modo a garantir que o produto final atinja os objetivos estabelecidos. Os diagramas UML, incluindo o caso de uso, diagramas de fluxo de dados ajudaram a ilustrar o fluxo de informações no sistema. Esses projetos, juntamente com os diagramas de classe, foram convertidos numa solução de trabalho durante a fase de desenvolvimento do sistema conforme planeado na fase de planeamento do projeto da metodologia ágil.

O sistema foi testado de modo a avaliar se as metas e objetivos do sistema foram atingidos conforme o estabelecido nos requisitos funcionais do sistema. A aferição dos objetivos definidos nos requisitos funcionais foi efetuada usando um caso de uso para teste. Para tal foram testadas todas as funcionalidades do sistema para de modo a identificar metas e objetivos não alcançados. Estas “imperfeições” foram comunicadas ao “desenvolvedor” do projeto para correção nas versões subsequentes. O sistema desenvolvido foi entregue à empresa XSealence S.A para integração com outra aplicação de desktop principal. Toda a manutenção e implantação do sistema será feita pela empresa. Para concluir o projeto foi elaborada documentação de modo a descrever e ilustrar todas as atividades do projeto.

Esta página foi intencionalmente deixada em branco

Index of Figures

Figure 1: XSealence business and technology partners	3
Figure 2: MSDN Weather Report in bar graph	8
Figure 3: MSDN line series graph	8
Figure 4: WMS Architecture	10
Figure 5: Example of WMS Request	12
Figure 6: Layers of Copernicus product.....	13
Figure 7: WMS Layers.....	13
Figure 8: Numeric data of weather	14
Figure 9: Marine traffic layers	15
Figure 10: Marine Traffic Numeric Representation	15
Figure 11: Wind layer of Predict Wind application	16
Figure 12: Fish Weather (FW) layers	16
Figure 13: GIS layers	20
Figure 14: Vector GIS.....	21
Figure 15: Raster GIS	21
Figure 16: OSM in Tatuk editor	23
Figure 17: OSM tiles with layers.....	24
Figure 18: Tatuk OSM tiles in windows form	25
Figure 19: WPF with OSM and Layers	26
Figure 20: Sharp Map with WMS layers	26
Figure 21: waterfall model of the software development model (SDLC)	29

Figure 22: Incremental Model of the software development model (SDLC) showing various increments up to nth increment	30
Figure 23: Agile Model of the software development model (SDLC).....	31
Figure 24: work breakdown structure activities was used to develop Gantt chart	36
Figure 25: The project Gantt chart	37
Figure 26: Earth observation desktop application workflow diagram	41
Figure 27: The system use case diagram.....	43
Figure 28: Earth Observation level 1 Data Flow Diagram	45
Figure 29: The desktop application class diagram	46
Figure 30: Weather data in.NC file	50
Figure 31: A bar graph showing how common is SQL injection is performed by attackers.....	53
Figure 32: Oracle database firewall showing its alert, allow and blocking boundary.....	54
Figure 33: Architecture of Oracle DB	54
Figure 34: Downloading data Architecture	55
Figure 35: Columns in Database Table	56
Figure 36: Task Scheduler in the windows server	58
Figure 37: Multiple Triggers	58
Figure 38: WMS Architecture of fetching layers	60
Figure 39: Copernicus Services.....	61
Figure 40: State of Sea window	62
Figure 41: Base map (all checkbox unchecked).....	62
Figure 42: Base with Layer temperature	63
Figure 43: Architecture of requesting the data	64
Figure 44: layers numeric data representation	65
Figure 45: Annotation representation	66
Figure 46: WPF window of numeric data	67
Figure 47: WPF window with Graph representation.....	68

List of Acronyms

Throughout this document several acronyms are presented, the meaning of which is described in the table below.

Sigla	Significado
VMS	Vessel Monitoring System
ICC	Integrated Control Center
MSDN	Microsoft Developer Network
OGC	Open Geospatial Consortium
WCS	Web Coverage Service
WFS	Web Feature Service
FW	Fish Weather
WPF	Windows Presentation Foundation
GIS	Geographical Information System
FAO	Food and Agriculture Organization of the United Nations
DEM	Digital Elevation Model
SDK	Software Development Kit
GMES	Global Monitoring for Environment and Security
ESA's	European Space Agency
EUMETSAT	Exploitation of Meteorological Satellites
EEA	European Environment Agency

HTTP	Hypertext Transfer Protocol
DFD	Data Flow Diagram
RDBMS	Relational Database Management System
IDE	Integrated Development Environment
XE	Express Edition
I/O	Input and Output
C#	C Sharp
MONICAP	MONItorização e Controlo das Atividades de Pesca
MVC	Model View Controller
UI	User Interface
OSM	OpenStreetMap

Esta página foi intencionalmente deixada em branco

Index

EARTH OBSERVATION	I
EARTH OBSERVATION	II
ACKNOWLEDGEMENT	III
ABSTRACT	V
RESUMO	VIII
INDEX OF FIGURES	XI
LIST OF ACRONYMS	XIII
INDEX	XVI
CHAPTER 1: INTRODUCTION	1
1.1 BACKGROUND INFORMATION.....	1
1.2 SIGNIFICANCE OF THE PROJECT	1
1.3 RECEPTION ENTITY.....	2
1.4 SIGNIFICANCE OF RESEARCH	3
1.5 THE PROPOSED SOLUTION	4
1.6 AIMS AND OBJECTIVES.....	5
1.7 STRUCTURE OF THE DOCUMENT	6
CHAPTER 2: STATE OF THE ART	7
2.1 OVERVIEW.....	7
2.2 WEATHER.....	7
2.3 WMS LAYERS	9
2.4 WMS ARCHITECTURE.....	10
2.5 EXISTING APPLICATION USING WMS.....	12
2.6 CONCLUSION FOR STATE OF ART.....	18
CHAPTER 3: GIS MAP ENGINES	19
3.1 OVERVIEW.....	19
3.2 GIS MAP ENGINES.....	19
3.3 TATUKGIS ENGINE.....	22
3.4 APPLYING LAYERS USING TATUK EDITOR	23
3.5 WINDOWS FORM	24
3.6 WPF (WINDOWS PRESENTATION FOUNDATION)	25
3.7 SHARP MAP.....	26
3.8 CONCLUSION FOR SELECTING GIS.....	27
CHAPTER 4: METHODOLOGY AND MANAGEMENT	28
4.1 OVERVIEW.....	28
4.2 PROJECT METHODOLOGY	28
4.3 PROJECT MANAGEMENT	34
4.4 CONCLUSION FOR PROJECT METHODOLOGY AND MANAGEMENT	37
CHAPTER 5: COPERNICUS (MARINE ENVIRONMENT MONITORING SERVICES)	38

5.1 OVERVIEW.....	38
5.2 COPERNICUS MARINE ENVIRONMENT	38
5.2.1 The Copernicus Space Component.....	38
5.2.2 The Copernicus InSitu component:	39
5.2.3 The Copernicus Service component	39
5.3 TECHNOLOGIES USED.....	40
5.4 WORKFLOW ARCHITECTURE OF BOTH FRONTEND AND BACKEND	41
5.5 USER CASE DIAGRAM	41
5.6 DATA FLOW DIAGRAM	44
5.7 CLASS DIAGRAM	45
5.8 CONCLUSION FOR SYSTEM DESIGN AND COPERNICUS	47
CHAPTER 6: BACKEND DEVELOPMENT	48
6.1 OVERVIEW.....	48
6.2 DOWNLOADING DATA FROM COPERNICUS	48
6.3 MOTU CLIENT	49
6.4 IMPLEMENTATION.....	49
6.5 ORACLE DATABASE	51
6.6 ARCHITECTURE OF ORACLE DB.....	54
6.7 ARCHITECTURE FOR DOWNLOADING THE DATA.....	55
6.8 IMPLEMENTATION.....	55
6.9 SCHEDULED SCRIPT ON THE SERVER.....	57
6.10 CONCLUSION FOR BACK-END DEVELOPEMENT.....	59
CHAPTER 7: FRONTEND DEVELOPMENT	60
7.1 OVERVIEW.....	60
7.2 THE ARCHITECTURE OF WMS LAYERS.....	60
7.3 APPLYING LAYERS	61
7.4 ABOUT THE LAYERS	61
7.5 REPRESENTATION OF WEATHER FORECAST DATA:	64
7.6 ARCHITECTURE OF REQUESTING THE DATA FROM THE DATABASE.....	64
7.7 IMPLEMENTATION WITH WINDOWS FORM	64
7.8 IMPLEMENTATION WITH ANNOTATION	66
7.9 IMPLEMENTATION WITH WPF WINDOW.....	67
7.10 SYSTEM TESTING	68
7.10.1 Unit Testing.....	69
7.10.2 White Box Testing.....	69
7.10.3 Quality Assurance Testing	69
7.10.4 Test Cases	70
7.11 CONCLUSION FOR FRONT-END DEVELOPMENT SYSTEM TESTING	75
CHAPTER 8: CONCLUSION AND FUTURE WORK	76
BIBLIOGRAPHY	78

Esta página foi intencionalmente deixada em branco

Esta página foi intencionalmente deixada em branco

Chapter 1: Introduction

1.1 Background information

The weather forecast is fundamental for conducting our daily activities. Agriculturists agree that weather forecast information helps them to prepare for the planting and for harvest. Airport control centers need to comprehend local environmental condition to schedule flights. Fishing activities might want to know the prevailing weather information before going into the sea for fishing. Weather forecasting help makes informed decisions on everyday life as it offers various information including expected rains, wind strength and direction, expected temperatures, humidity, atmospheric pressure and many other aspects of weather.

Weather forecasting includes collection available weather information describing the current state of the atmosphere (particularly the temperature, humidity, and also the wind) and using the understanding of atmospheric processes (through meteorology) to determine or predict how the atmosphere evolves within the future.

Present day climate estimating infrastructure includes a blend of computer models, observation, and a knowledge of trends and patterns. Utilizing these strategies, sensibly precise estimates can be made up to around one or two days ahead of time. From this data, point by point estimates is less helpful, since atmospheric conditions like temperature and wind direction are extremely unpredictable.

The vast majority of the computer models utilized for forecasting are controlled by the National Weather Service, which makes figure models in view of complex equations. These models are utilized by a wide range of climate and news benefits in getting ready day by day estimates. Local climate observers, balloons, satellites, and climate stations additionally help give information on weather derived from the collected data estimates.

1.2 Significance of the project

There is increasing recognition that climate encompasses a central role in world economic and social sectors. Climate directly affects the prosperity within of the many sectors, like fishing, insurance, agriculture, energy, and health. Similarly, unfavorable weather conditions have

negative and prolonged impacts, as well as reduced private sector investment and productivity related to economic and environmental uncertainties.

Weather forecasts play a very important role in the fishing activities because they to plan their schedules by considering the weather forecasts. And in the recent days fishing activities are increasing day by day, and overfishing is one of the major problems in the marine territory which is grabbing the attention of the authorities of national and international. Overfishing only occurs, when the number of fish caught exceeds the set number of fishes to be caught in a certain period of time and there is no natural reproduction replacement done[1]. Overfishing creates marine life imbalance as well as affect the coastal communities who depend on fish for their way of life. Billions of people depend on the fish for proteins, in the case of marine life imbalance if there is an imbalance then it's going to affect the billions of people around the world. In today's life, more than 85 percent of fisher's life have been crossed beyond their biological limits and also, they are targeting the particular species in the ocean which leads abundance of smaller marine species for this it requires strict rules and regulations.

The rate of illegal fishing activities has been increasing in the recent years in various areas. Various fishing regulation authorities have tried to come up with policies, procedures, and systems to curb illegal fishing. As a result, various technologies to avoid illegal fishing catches are been developed and implemented. Such regulation bodies include the XSealence, which is one of the companies which are developing vessel tracking application. The application allows regulation bodies can keep track of the vessel. The application monitors fishing activities and can detect when illegal fishing catches have occurred as well as a monitor if captured fish belong to protected species. The application also helps determine if the limit of weight of the allowed species has been exceeded. By using these kind technologies, we can easily maintain the marine life in a balanced way.

1.3 Reception Entity

The entity hosting this stage was Xsealence S.A. - Technologies do Mar SA, a company founded in 2013. Its main business area is maritime technologies, namely control, monitoring, and control. XSealence belongs to the Trailtec group to which Tecmic also belongs.

XSealence has roped in various partners so as to increase its service delivery and ensure quality marine life is protected. These partners include Aqua fish, JRC, IRPA, COMSAT, IRIDIUM, INMARSAT, NAUTEMA, Grupo LR of Israel, CE among other as shown in the figure below. XSealence relies on these entities for enhanced performance. The companies

share expertise and ideas on how to improve marine life. At the client level, it has MPA, IRPA, DGRM, among others. The area of influence extends to Portugal, Angola, Turkey, Ireland, Spain, and France.



Figure 1: XSealence business and technology partners

1.4 Significance of Research

In the past years, fishing was thought to be a leisure activity which mostly was done by the youth and their hobby. However, today fishing is considered a commercial activity which has attracted a lot of interest and activity alike. Fishing has grown to be a source of living for various families. Countries neighboring seas and oceans have engaged commercial fishing with some countries exporting the fish. This serves as part of the country income and thus boosting their economy. With commercial fishing being carried out in various water bodies, fish regulation and monitoring have grown to be one of the most key activity in marine life protection.

Various technologies are emerged to monitor shoals and fishing vessels which can carry tons of fish. This evolution technology has allowed increasing the fish supply as well as allow capturing of different variety of fish species and deliver them to the market. This kind of development has brought good things for humanity, upon that it has also brought the problem of overfishing and illegal catches.

In the last few years back there were no problems like overfishing because at that time there were no technologies and no proper vessels to carry the fish. These problems have gained increasing attention in recent years as they challenge sustainable fishing. Overfishing is not a recent phenomenon but has been there and its worsening in recent decades[2]. According to the FAO, today, 25 percent of all the world fish stocks are either over-exploited or exhausted and another 52 percent are being fully exploited. Fish overexploitation has led invited slow and sure collapse of the fish industry with almost 80 percent of the fish and their species being overexploited, depleted or in the state of collapse[3]. Overfishing that can be described and characterized in several ways, but the idea is almost always the same: to make excessive catches in a short period in each ecosystem. In this way, the ecosystem will gradually degrade, for this reason, the quantity and variety of species present in this ecosystem are becoming smaller and smaller [4]. Illegal fishing creates a number of problems in the marine world, and it's made to create the measures on them, such as to keep track the catches that are made. This Illegal catch is also a problem for sustainable fishing. This kind of overfishing should have rules and regulations in order to maintain the ecosystems in a balanced way and if they cross the limit they should be punished.

1.5 The proposed solution

The proposed system is a desktop application that gives marine weather prediction for the last 10 days. The given data include water and ice speed, the direction of the wind and other aspects of marine weather. Nowadays sea technologies are improving day by day, in order to help the marine community people. XSealence is one of the sea technologies company which is developing the application called vessel tracking, previously they did it web application called "web vessel tracking", now they are working on the desktop application. XSealence already developed a device called MONICAP(monitors and control of fishing activity)[5] is one of the systems which can help to prevent illegal, overfishing and also keep track the vessel. This MONICAP system is a VMS (Vessel Monitoring System). In brief about these two concepts are, they are the systems which are installed in the vessel this system helps to monitor their position and realized catches. This information is then transmitted to a control center where the fishing activity is monitored by the competent authorities, in particular fisheries authorities and the maritime authorities [6]. By using these kinds of application, the user can see the information about the vessel, they can create alternative sea routes when the intended route has bad conditions as shown by the application. Also, the application has a good Integrated Control Center (ICC) which is the desktop application that is installed

together with the MONICAP software at the control center to communicate with the shipping vessel. In that project, my part of work is Earth Observation which is explained in detail in the upcoming chapters. The project was started so to help people around the globe see various weather information of some specific areas defined by their coordinates. The desktop application will show various marine climatic conditions such as ocean mixed layer thickness based on sigma theta, sea ice thickness, sea area fraction, ice velocity towards a various direction, potential temperature, water salinity, water velocity and other various sea weather aspects. This information will not only help the fishermen but sea sailors as well. The boat controllers as well as shippers control towers will be able to monitor prevailing conditions in the sea and about the locations of their ships and advice the ship crew accordingly.

Ocean or sea weather data is crucial for all sea users as it serves to predict the conditions to expect and prepare a disaster management plan for any eventuality. Such data can help save lives and help fishermen in their daily fishing activities. The information is equally useful to boat and ship users as they sail through the sea or ocean.

1.6 Aims and Objectives

The application was developed primarily to give weather forecast to sea and ocean bodies users:

- i. To provide marine weather forecast to fishermen and sailor as well as marine regulatory bodies;
- ii. To predict possible marine weather information such as ice and water velocity
- iii. To offer specific marine information about a specific location indicated by its coordinates;
- iv. To provide past 10 days data on marine weather to help in informed decision making;
- v. To design and implement an offline marine weather forecast information that can be installed on sea vessels to help in times of bad internet connection problems. The desktop application will offer reliable and easily accessible marine weather forecasts.

1.7 Structure of the document

This paper will be organized into various sections in form of chapters. Each chapter will seek to address a specific issue on fishing and weather monitoring of the proposed desktop application.

- Chapter 2: This chapter discusses the various systems that exist for vessel and fisheries monitoring.
- Chapter 3: This chapter discusses the GIS map engines.
- Chapter 4: This chapter explains in detail the development methodology used, project planning and management.
- Chapter 5: This chapter outlines the Copernicus Services and technologies used.
- Chapter 6: This chapter discusses the Backend development.
- Chapter 7: This chapter explains the Frontend development and Tests.
- Chapter 8: This chapter reflects on the work developed throughout this stage and describes the future work.

Chapter 2: State of the Art

2.1 Overview

In this chapter aims to describe the existing systems technologies that are used for the weather forecasting, and WMS (Web Map Service).

2.2 Weather

Weather provides the state of the sea based on marine data collected. For fishing activities, weather information is extremely important. Fishermen and fishing companies need ocean weather forecasts like sea temperature, sea water velocity, Ice speed, wind speed and so on, so as to know when it is safe to go fishing and when it is not. Earth consists mainly two important pressured environments weather (Air and Water). Sensible weather and the day-to-day weather that we experience every day is one of many factors that affect fishing as well as our daily lives. But weather plays a very important role for fishing because before entering the sea, the fishers will check the state of sea i.e. sea temperature, water velocity, the wind etc. By doing this they will get information about the state of the sea, and moreover, they can watch the earlier day's atmosphere and anticipate the future day's climate, and ultimately, they will get a sensible thought with respect to the weather.

For the weather forecasting, there are existing applications, which may be seen everywhere from desktop to smartphones or even wearable devices. some of them are Microsoft desktop application, Google weather, now a day there are so many weather applications are available. In the below figure2(which is taken from the MSDN application[7]) which gives the overview bar graph and also line series graph.

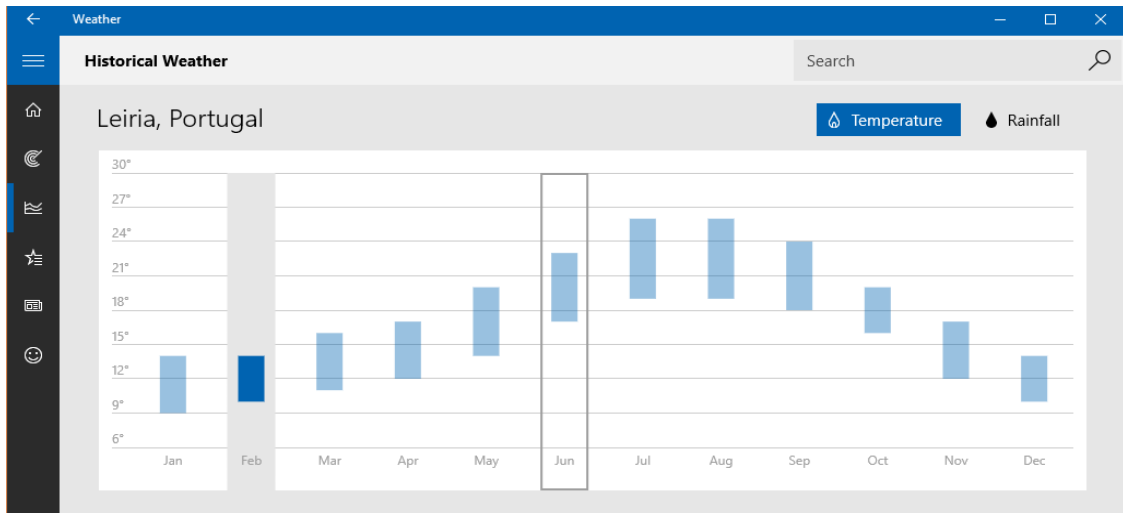


Figure 2: MSDN Weather Report in bar graph

In figure 2, we are observing two tabs in the i.e. temperature and rainfall. In the temperature it will give the normal temperature level of all months in a year, we can see that in the above figure we normal temperature of all months, however, this is not precise, it's just with an average temperature of a specific month.

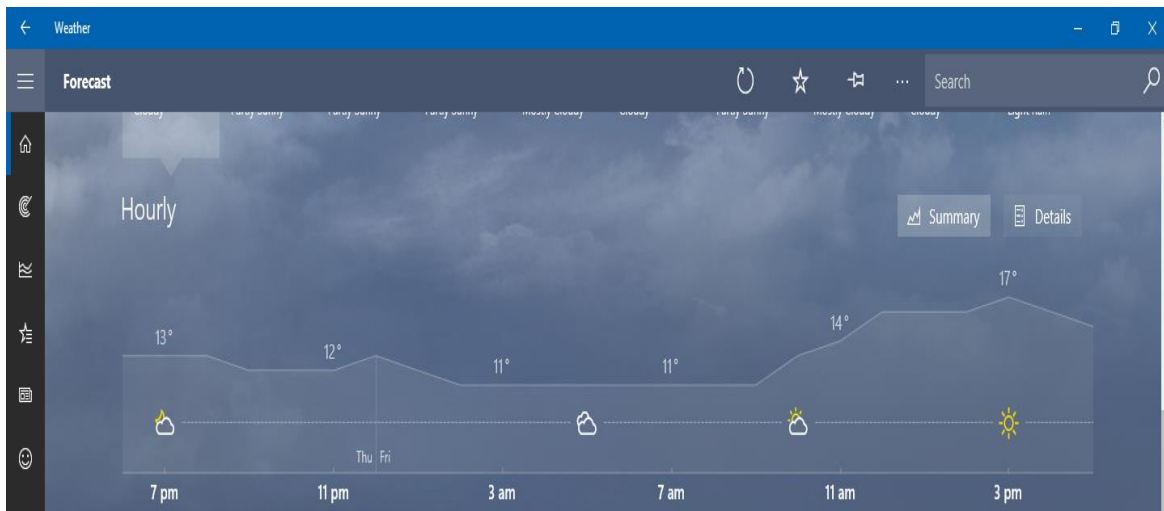


Figure 3: MSDN line series graph

In figure 3, its line series graph but this is giving the information on hourly basis, this hourly basis temperature is also important for fishing because they can easily decide about the next movement.

For fishing activities with these weather data and graph if its include the layers like temperature, wind, water velocity, ice speed etc. on the top of the map it's easy to recognize the weather information. In the next section is going to discuss these WMS layers.

2.3 WMS Layers

In the recent last ten years or so, there has been a great innovation of geographical information science and technologies which have tremendously increased and thus making the whole idea of marine sea weather forecasting easy and enjoyable. Today there are more than ten thousand map layers are available in online through Open Geospatial Consortium (OGC) specified an interface, such as Web Map Service (WMS), Web Feature Service (WFS), and Web Coverage Service (WCS)[8]. A Web Map Service (WMS) layer is built on data from OGC (open geospatial consortium) WMS server. The web map service layer allows the users to access these services over the internet. It produces maps of spatially referenced data dynamically from geographic information[9].

WMS provides the spatial information to the internet users by the map images, this spatial information or data are stored as vector data. It is very difficult to load the whole image showing the entire map of a requested location, thus when the requests are made, the image is divided into several sub-rectangles. Subdivision to multiple rectangles allows fast to loading of the images on the map, for example each sub-rectangle is sent to a WMS node to produce some sub-maps, the resulting map is produced by these sub-maps and moreover all these sub-maps are produced in parallel so in this way its saves the loading time of the WMS layers on the map [10].

2.4 WMS Architecture

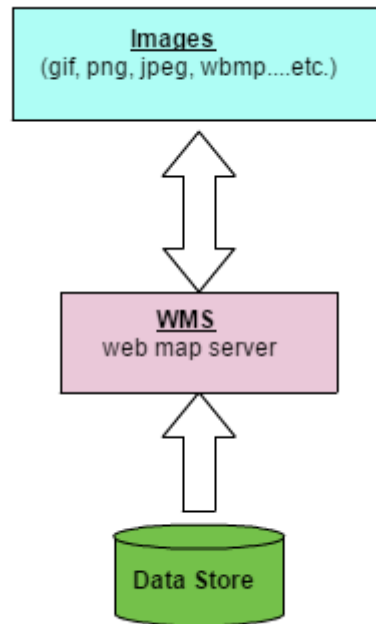


Figure 4: WMS Architecture

The above figure demonstrates the WMS architecture, the data is stored in the data store and it's made accessible to others for use by means of the WMS. The more details about the WMS is explained in the upcoming chapters.

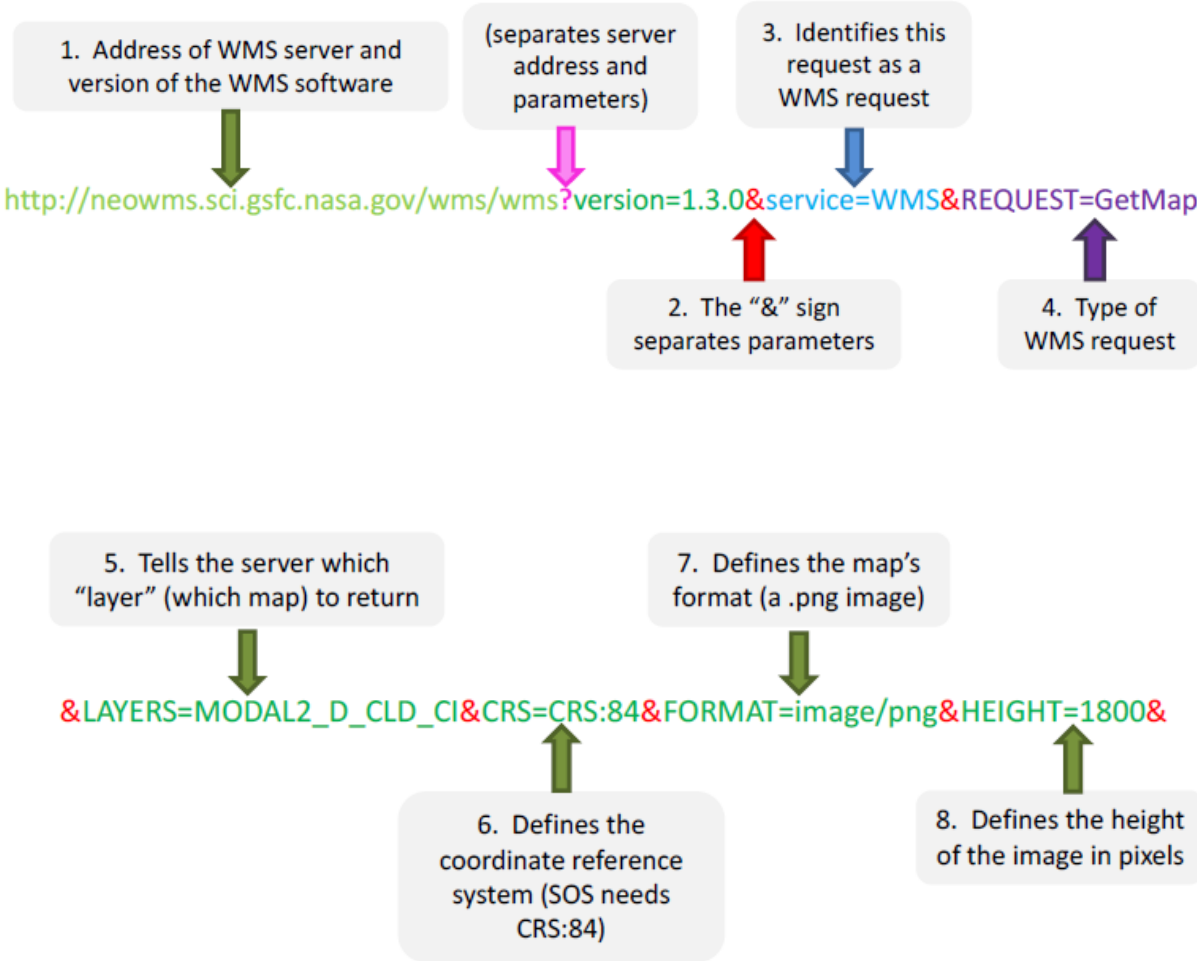
A simple WMS request is explained as follows:

1. Server details (a WMS 1.3.0 request);
2. Request type (WMS Get Capabilities);
3. Layer name (global-analysis-forecast).

The WMS provides simple HTTP interface for requesting geo-registered map images from one or more distributed geospatial. The WMS requests define the layers to be processed, the response to the request is in the form of images (returned as JPEG, PNG, etc.) which is displayed on the desktop or browser application. The request and response from the WMS are done according to a specific standard[11], there is three standard and third is optional from the given below:

1. GetCapabilities: This standard, which can query the WMS' capabilities;
2. GetMap: This standard, requests a specific map;
3. GetFeatureInfo: This standard, requests information on individual map objects.

WMS solicitations are composed of the address of the server the request is being sent to and a progression of parameters. There are two sorts of request that are needed for any WMS request for GetMap and GetCapabilities. In spite of the evidence that GetCapabilities is typically the main request sent, it's less complex than GetMap and not generally required, so it's clarified later. A GetMap ask for is diagrammed beneath, separated into its segments. The GetMap ask is broken into three lines, however, it's only one demand. GetCapabilities solicitations can be made by embedding's "GetCapabilities" set up of "GetMap" in the "REQUEST=" parameter. No different parameters should be indicated.[12] For instance, the valid GetCapabilities request is as given below diagram.



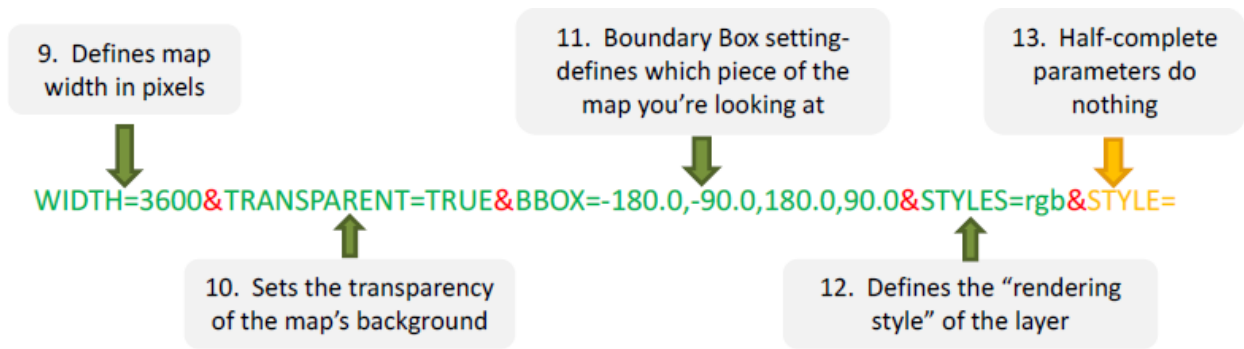


Figure 5: Example of WMS Request

In the above figure, it's clearly explained that:

1. Server address which is provided in every WMS request made. The server address ends with question mark parameter;
2. The sign & Separates statements/parameters;
3. Defines this string of letters as a WMS request;
4. WMS request defines the type of WMS request;
5. Additional Parameters.

In the next section going to explain the existing application which are using the WMS layers.

2.5 Existing application using WMS

In the present days, there are many applications which are using the WMS service, it may be a desktop application or Web application. The WMS services are very important to the marine traffic control and also human activities, by the help of the WMS services, weather layer can apply on the top of the map so people can easily recognize where is it's more or less by seeing the WMS layer. The reference application for our desktop application is Copernicus product, the Copernicus is the European Commission site. Copernicus is a European Commission site which can provide the weather data, WMS services and so on for the marine environment and monitoring. And it's having their own product which used WMS services for displaying the layers and it also shows the numeric data representation as shown in the below fig(taken from Copernicus product)[13]

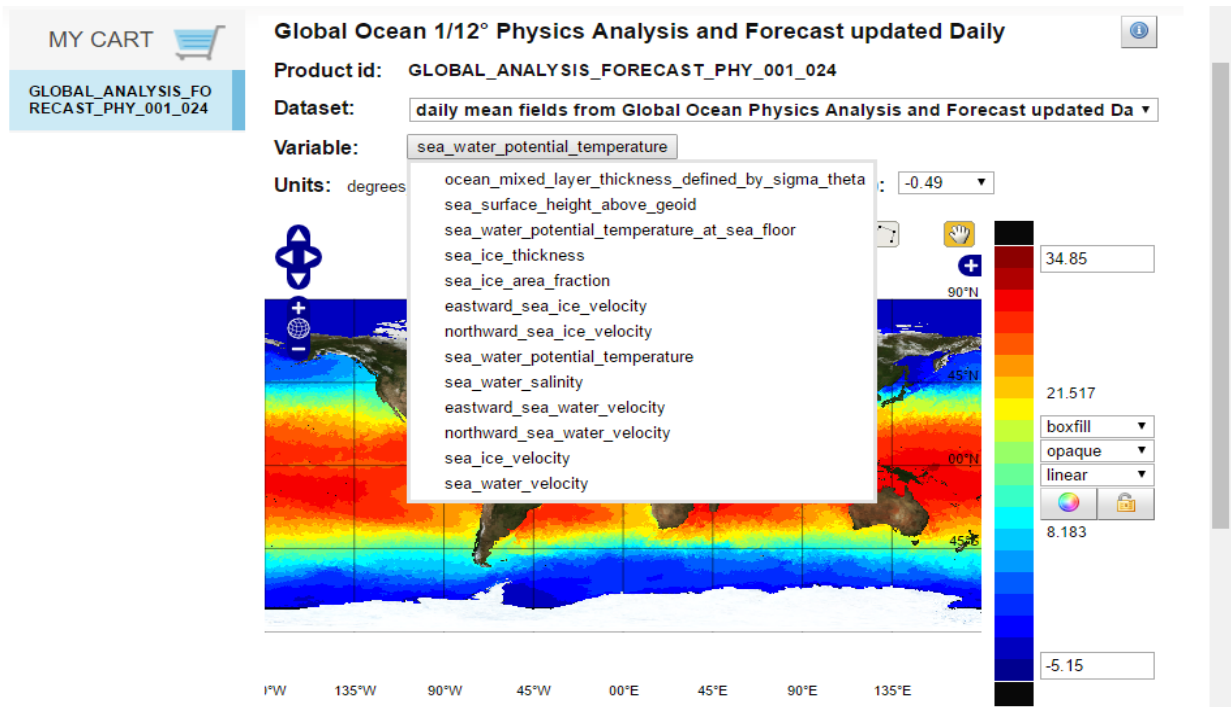


Figure 6: Layers of Copernicus product

In the above figure 6 we are seeing various weather layers like temperature, sea surface height, water velocity, ice velocity, potential temperature at sea floor, salinity etc. when the user selects the layer, its connects to the server as discussed in the above section in order to reflect to the other layers on top of the map which is as shown in the below figure.

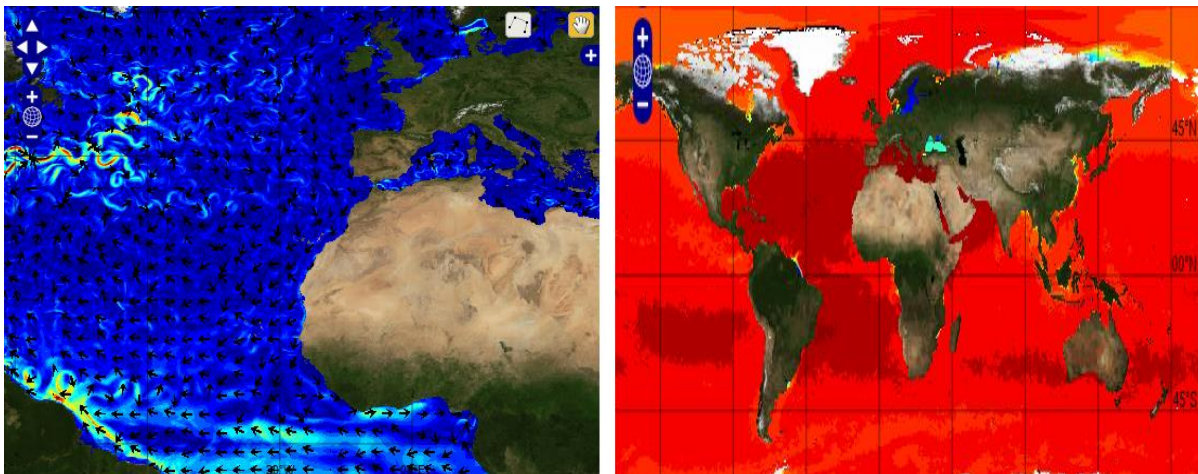


Figure 7: WMS Layers

The numeric information of weather is essential for the marine monitoring because by examining the past estimations of the weather information, they can without much of a stretch predict the following day esteem. Copernicus website represents weather information using

numeric data. The user clicks at a point on the map, for instance when users click at a particular latitude and longitude it's going to give the value of that specific point.

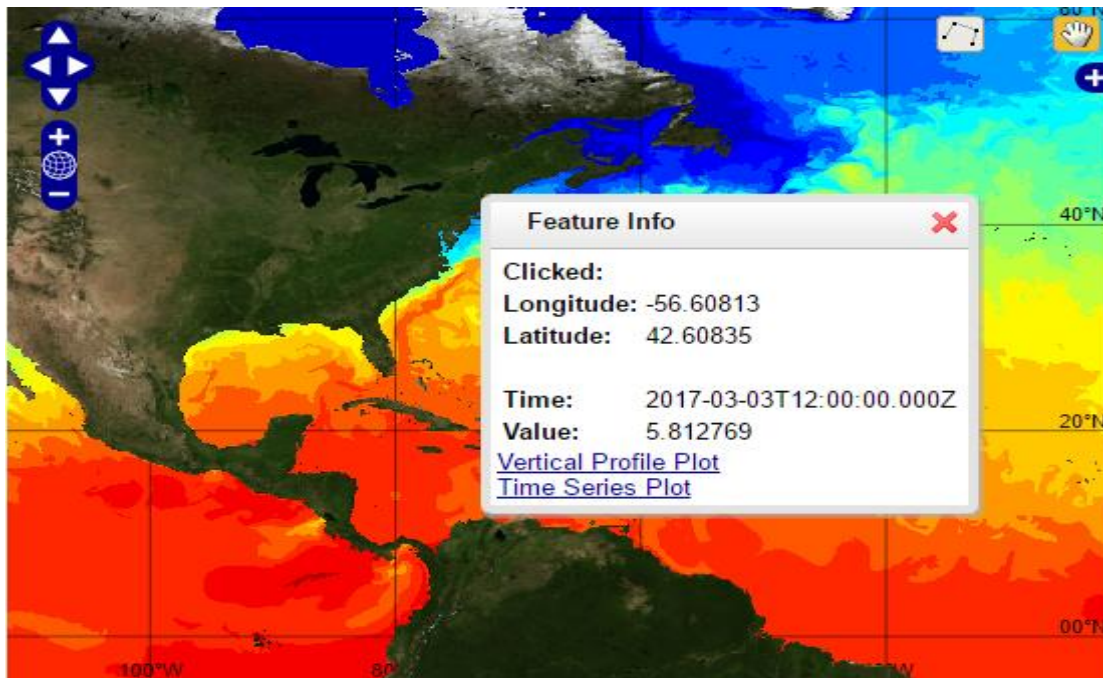


Figure 8: Numeric data of weather

In today's life, there are many desktops applications like Predict Wind Offshore, MSN weather etc. and web application like marine traffic, fish weather (FW) etc. Predict Wind runs a global weather model of 50-kilometer resolution using two sources. It generates 1 kilometer or 8-kilometer resolution forecasts for world's popular areas [14]. Predict Wind own several mobile applications and forecasting website as well to assist interpret weather data collected. It uses various tools and representation charts such as line, bar graphs as well as wind arrows and maps to show weather information.

MSN weather runs weather forecast mobile apps and the web an online web application which offers the latest conditions expressed in hours, days or on the 10-day scale. The weather app was designed to run on Windows platform. It supports various data representation such as images and graphs. The app's features access a 10-day and hour forecast from various weather data providers and historic weather averages. This capability gives it a current weather data about various cities over the globe. The data details the weather data, Doppler, cloud cover, travel maps as well as satellite. It helps prepare for the harsh and severe weather in any city.

Marine traffic[15] is one of most useful application for the vessel tracking in this application they are providing the various layers on the top of the map, but they representing only one weather layer i.e. wind layer which is as shown In the below figure 9(which is taken from marine traffic[15]).

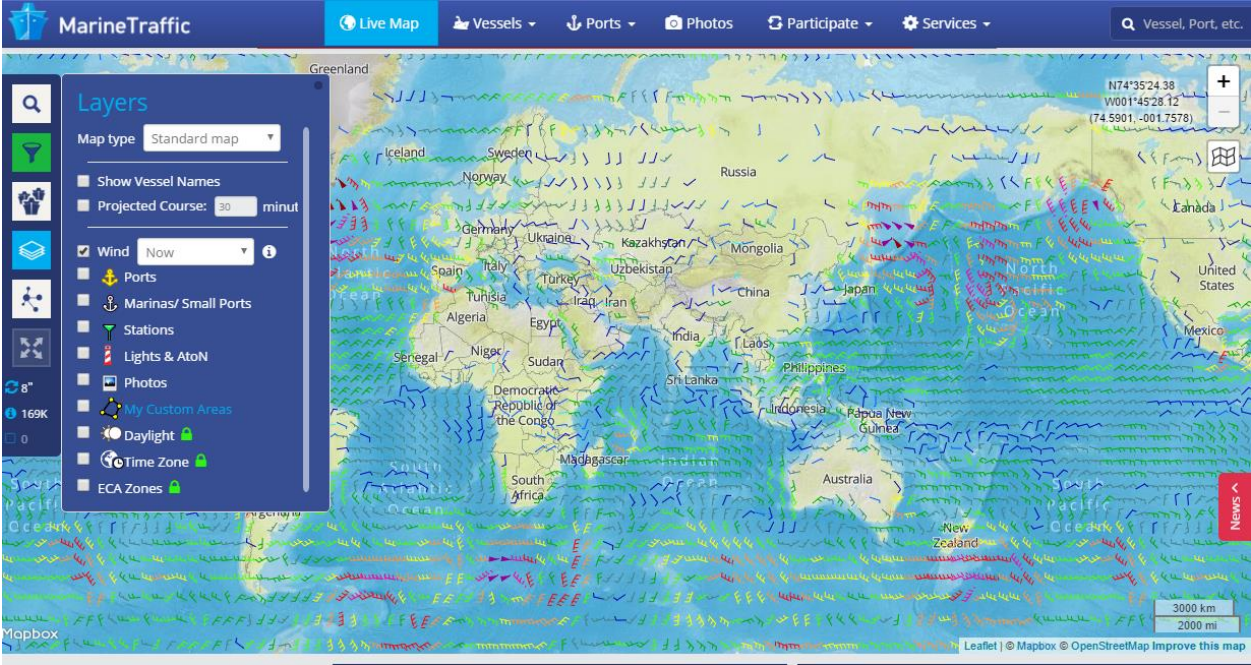


Figure 9: Marine traffic layers

In the below its showing the numeric data representation of user clicked a point on the map.

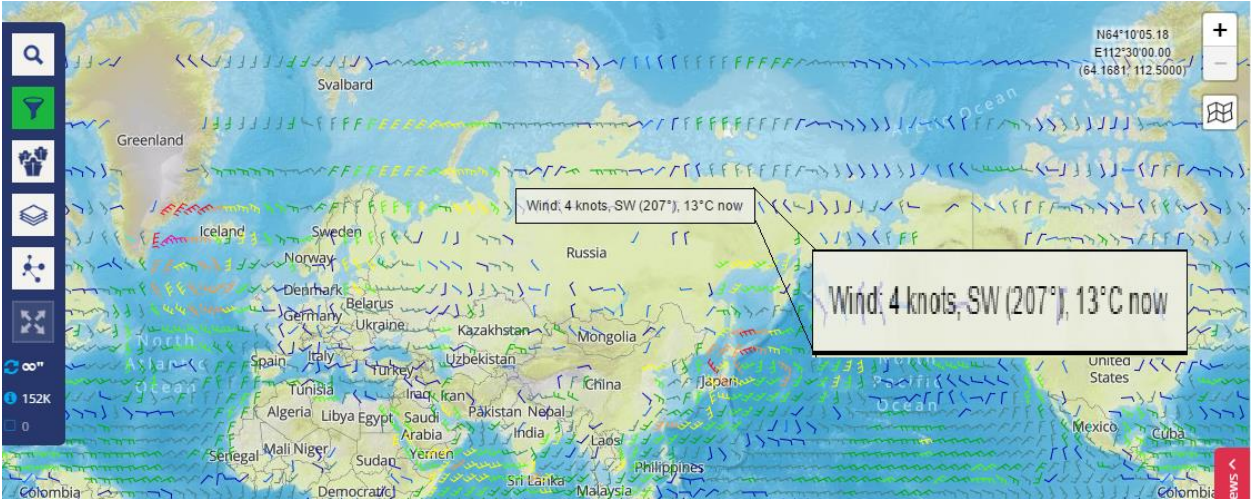


Figure 10: Marine Traffic Numeric Representation

In the present technical world, there are many desktop applications related to marine monitoring, Predict Wind Offshore is one among them. This one of the good desktop application which can predict the wind, in this application they display the flow of wind

direction, the live wind direction is very helpful for the users or the marine community people. Its gives alerts to the alerts to the users and represents the wind flow direction in arrows and also in wind bars as shown in figure 11 [16].

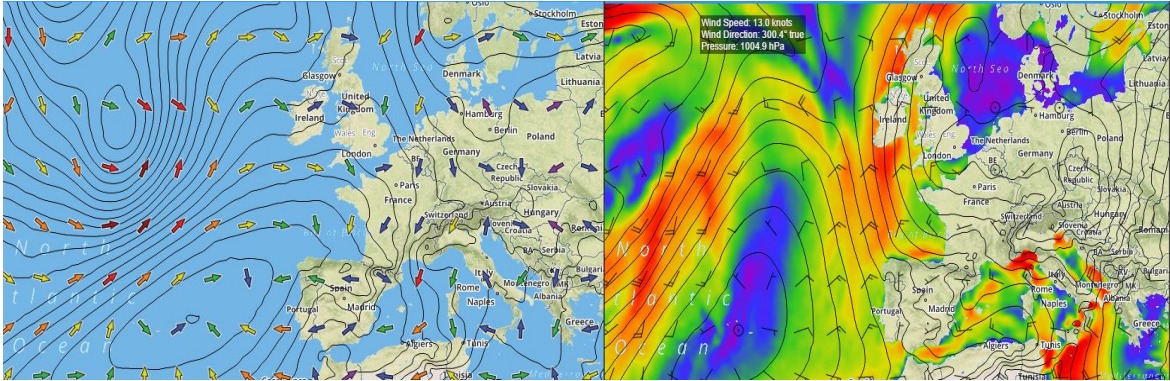


Figure 11: Wind layer of Predict Wind application

Fish weather (FW) is another application which helps for fishing activities, it can predict the weather forecasts by applying the layer on top of the map, in this application, they present various layers of weather information like temperature, wind, clouds etc. For fishing activities, the weather forecast is very important because by seeing this layer they can easily decide the fishing activities. The application displaying the weather layers is as shown below[17].

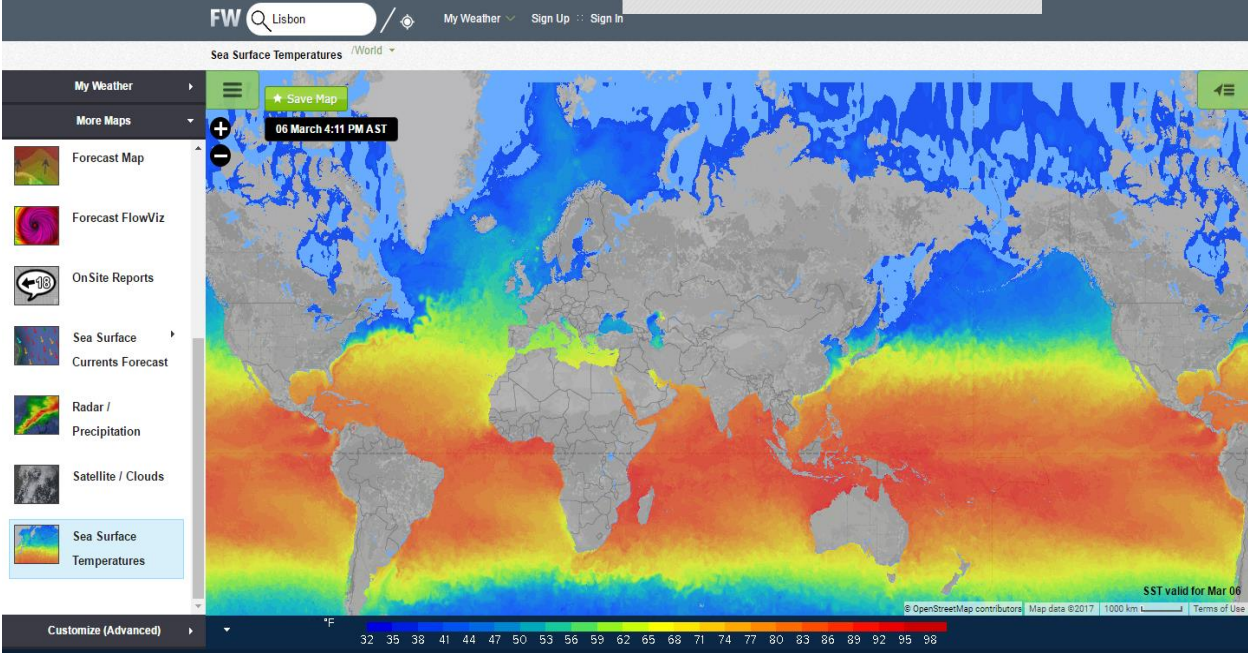


Figure 12: Fish Weather (FW) layers

Most of the weather apps discussed either run on windows or android applications. However, Linux, iOS platform users have been equipped with weather applications as well. Weather underground and free iOS app have entered the weather forecast industry. The app runs on both iOS and Android devices. Unlike other apps, Underground weather app uses a network of personal weather stations. Other apps usually use NEXRAD radar data. The weather stations have weather measuring suite of instruments. There are over 200,000 personal weather stations that gauge temperature, wind speed, humidity, pressure, and direction [18]. The app gives a super detailed weather forecast hourly. The give radar imaging as well as projected storm path. Weather Underground is user oriented as gives you what you really need. For instance, most apps give when the sun rises and set but gives information on when it will get dark. This helps travelers decide if to cycle faster so as to reach the expected destination before it gets dark.

Dark Sky an Android and iOS app pulls data from national oceanic and atmospheric administration network. The network consists of 140 radar stations spread across the US using some fancy math to predict the hyper-local weather. International dark sky association believes weather can be chaotic and individuals can see and predict weather patterns in local and small scale. The app uses phone sensors to collect and generate local weather forecast via its report feature on NOAA radars. The app focuses on now and presents location data. The data predicts the next hour and within the user area[18]. The app has the ability to tell which minute of the next hour it is going to rain through extended forecast features. The app also has slick global view feature for global radar which show broad weather patterns over the globe. Users can zoom in and see their country, state, and town.

Rain Aware is a hyper-local short-term forecasting feature referred to as “nowcasting”. Rain Aware extracts data from radar, satellites and surface stations. The app has ability forecast precipitation in the user local area as a way to predict the local storm path. The app gives a detailed view of the weather in the next 3 hours. The app tells when is going to rain and how long it will rain. The app also gives a general prediction of the next 24 hours, as well as 7 days, predict with up to date images[18]. The up to date imagery give minute storm information and constantly update the prediction based on the storm movement. The app has the capability to tell the nearest town where rain if falling

2.6 Conclusion for State OF Art

In the above sections discussed about the existing application. There are various apps have been developed to predict marine weather. Some of the apps such as Rain Aware have real time reporting systems. However, the systems concentrate on only one element of weather such as rain, storm, wind or humidity. Today many applications developed on weather forecasting run on android, and iOS phones as the data carried is only useful to individuals. However, earth observation is the part of the application that shows various voluminous marine data which include temperature, water salinity, water speed, ice velocity and well as wind direction and speed. The application can be installed on marine vessels to give weather predictions for the last 10 days which other apps do not give.

In the upcoming chapters is going to discuss the technologies used and implementation of the project in detail.

Chapter 3: GIS Map Engines

3.1 Overview

This chapter discusses the two GIS search engines namely the Tatuk GIS and Sharp map. The first section discusses the Tatuk GIS in the Tatuk editor, on the top of the Tatuk OSM applying the weather forecasts layer. The discussion Visual studio that is window form and WPF (Windows Presentation Foundation), later discussed the sharp map and final conclusion about the two GIS search engines.

3.2 GIS Map Engines

GIS stands for "Geographical Information System" which is a framework for capturing, storing, checking. GIS is primarily concerned with a showing of the related information to positions on Earth's surface. GIS is the blend of computer both hardware and software as talked about in the above explanation, it is composed around advanced database which can store, manipulate, capture, analyze, create, and display spatially referenced data [17].

GIS employs the use of polygon warning to perform the geospatial analysis. GIS is used to develop the proposed application as it will help in collecting the weather data on various locations identified by their coordinates. The data will be used for plotting, analysis, visualization and weather data interpretation. The interpreted data will serve as the prognostication of weather. The application will help the marine authorities and users in instantaneous of determining weather data in specific areas indicated by use of longitude and latitude at various atmospheric isobars [19]. The GIS

GIS empowers people more effortlessly observe, analyze, and understand patterns and relationships with GIS technology. GIS users can compare the locations of different things to discover how they relate to each other. For instance, by utilizing the GIS, the client can include a site that produces pollution like water contamination. This means using GIS, users can easily determine which wetlands are most at risk[19]. Indeed, even the client can apply the weather forecast layers on the top base map, which is clarified in detail in the upcoming chapter.

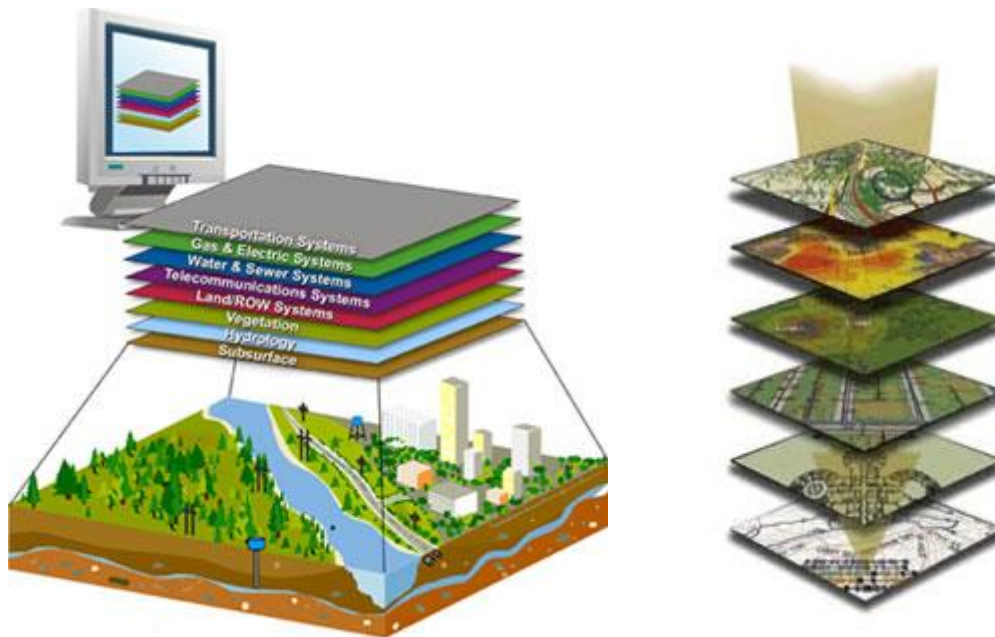


Figure 13: GIS layers

The different layers of map data are stored in the database, these different layers of the map are having its own layer of map theme, for example, a layer for roads, a layer of soils, a layer for rivers, etc. In figure 13(which is taken from [20][21]) is having a different set of layers, the GIS is a kind of electronic overlay, query, manipulate, and combine the various layers to create maps aimed at a specific purpose.

The map information is stored in the database is isolated into two fundamentals format which is the raster design and the vector principal configuration. The vector format can store spatial information as focuses, lines, or polygons. The vector GIS organization is much similar to conventional paper maps. The vector GSI in fig 14 [20] is having great database availability, great cartographic yield, yet it has poor surface displaying and restricted investigative abilities. Fig 14 demonstrates the distinctive components in a vector GIS, with soils spoken to as polygons, wastes as lines, and archeological locales as focuses.

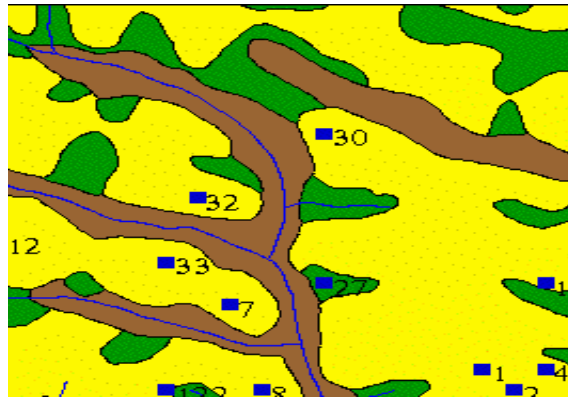


Figure 14: Vector GIS

The second method of GIS data modeling is raster. The model resembles spreadsheet arrange with lines and sections. GIS raster model has a great surface display with great logical abilities. However, it has to have poor cartographic yield[20]. The lines and segments of the spreadsheet are fixing to x, y directions of the crossing point of each line and section shaping a cell which compares to a particular region in this present reality. Fig 15 demonstrates a similar soil map in Fig 13, but now in a raster arrange.

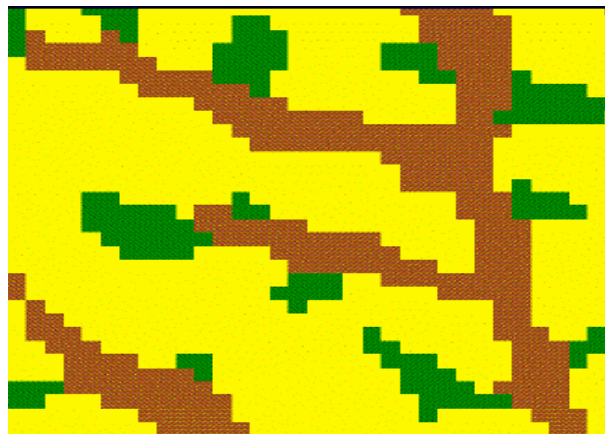


Figure 15: Raster GIS

The one very important type in the raster GIS is DEM (Digital Elevation Model). The changes in elevation in the raster format are indicated by changing z-axis values in each cell. The evolution in the DEM is continuously changing across the surface by creating a more realistic model[20].

3.3 TatukGIS Engine

The TatukGIS is one of the custom developing kits, its developer kernel is a professional-level GIS software development kit (SDK) for creating stand-alone, client-server, web server, embedded, and mobile applications. TatukGIS is useful for adding GIS functionality to effectively existing items, porting software from less creating custom GIS applications appealing GIS advancements, and from scratch [22]. TatukGIS product supports the multiple development platforms, some of them are Microsoft.NET for WinForms and WPF, Microsoft ActiveX/OCX, Microsoft ASP.NET (for web solutions), Microsoft.NET Compact Framework (for mobile solutions), Embarcadero Delphi/C++Builder VC etc. The best thing in the Tatuk GIS is that each developer kernel edition reflects essentially the same API, framework, and technology, which makes as possible to port GIS applications across development technologies and operating systems. The TatukGIS Developer Kernel is known for execution, rich and legitimate API, local support for most (raster, vector, and geodatabase) formats, state-of-the-art support for coordinate systems, 3D representation, consistency with many open principles, and sensible authorizing[22].

TatukGIS has the wonderful editor, Tatuk editor helps to merge the layers in an easy way, and again it has lots of options to do various kinds of maps. The TatukGIS Editor is a professional, universally useful desktop GIS mapping and information altering application with inherent scripting condition for customization and highlight expansions. The highlights include Viewing, 3D Viewing, Editing, Coordinate Systems, Rectification, Scripting, Databases. Locally (without import or configuration transformation) opens and shows an extensive variety of vector, picture, grid, and SQL database layer groups, including progressed spatial server geodatabases. WMS, WFS, and WMTS mapping services might be opened from the web. Complete visual layer property, legend, and scale controls support deep customization of the map appearance[23]. The outlook of the Tatuk editor is as shown in fig 16. The Open Street Map tiles of the Tatuk GIS in the Tatuk editor, its opens smoothly and clear it's as shown below[24]

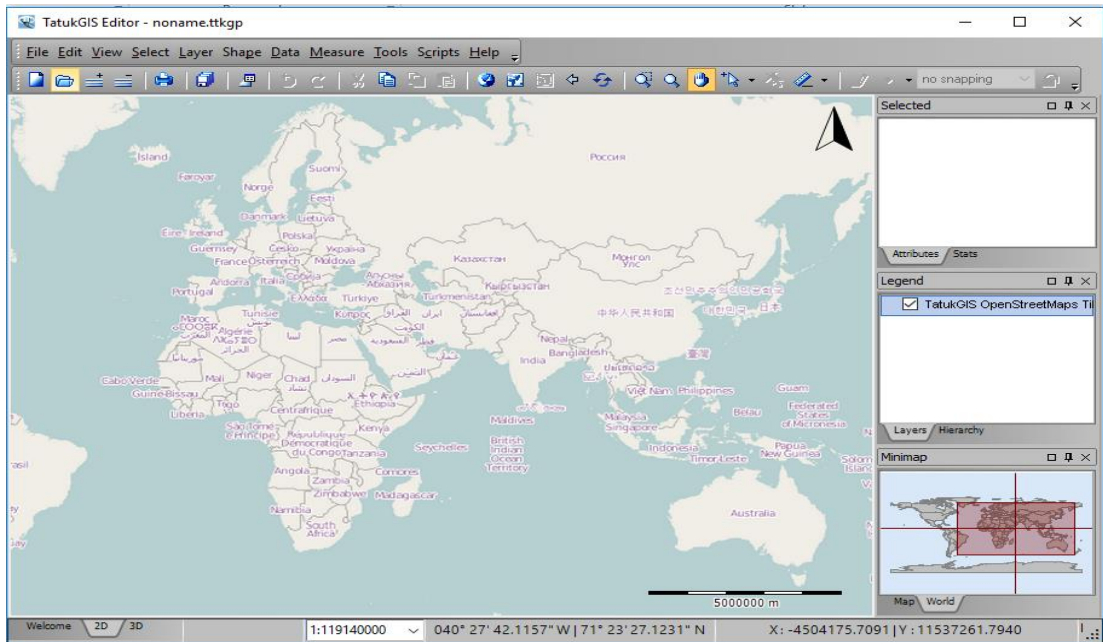


Figure 16: OSM in Tatuk editor

3.4 Applying layers using Tatuk Editor

Tatuk editorial manager gives the best choice to check the WMS service layers fits for the Tatuk OSM (open street map). Adding the layers to the OSM of the Tatuk GIS, from the Copernicus, is a WMS service (Web Map Service). In the client-side, they can have diverse variables like temperature, sea water velocity, sea ice velocity and so on. In this picture, the OSM tiles of the Tatuk GIS is the base of the map and the layers are on the top, the users' needs to make this service layer as translucent then both OSM tiles and layers are visible legitimately. It works fine in the Tatuk Editor, then for the testing purpose, it was implemented in Windows form as well as in WPF (Windows Presentation Foundation) which is shown in the next section.

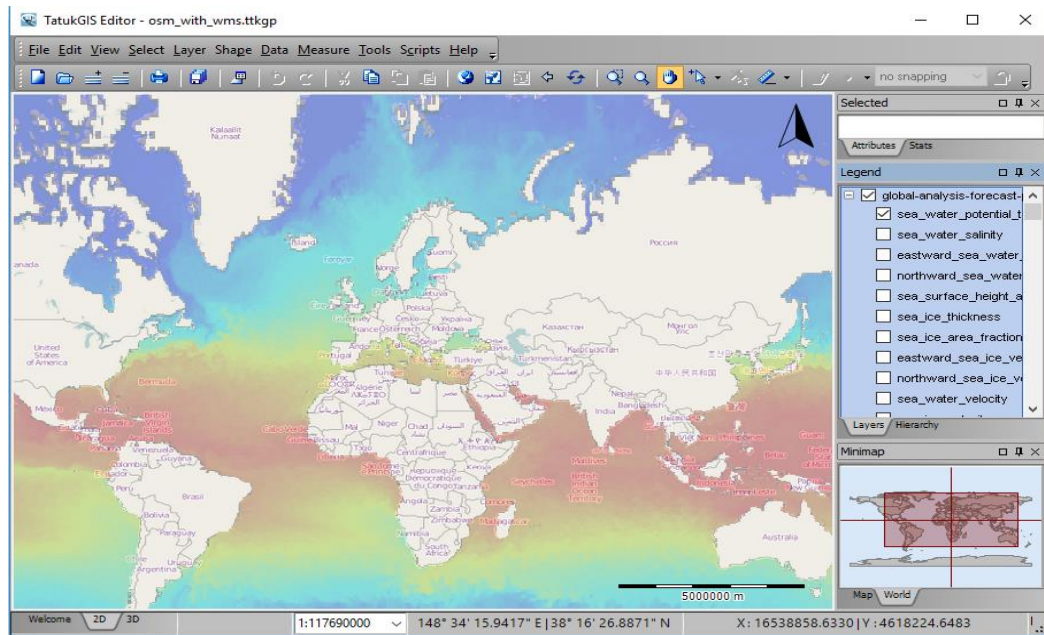


Figure 17: OSM tiles with layers

3.5 Windows Form

The earth observation desktop application was developed in various releases. Each release was closed tested by my supervisor and the feedback worked on. In the end of the project, we ended up with a WPF form of weather data representation which was perfected from annotation representation. In the windows form, OSM tiles of the Tatum GIS work properly and smoothly, but when the users try to add the layers on top of the OSM tiles of the base map, the OSM is not visible properly, its little bit hard to recognize the weather forecast layers, we overcome this problem in the WPF which is in the next section, the OSM tiles in the Windows form is shown below.

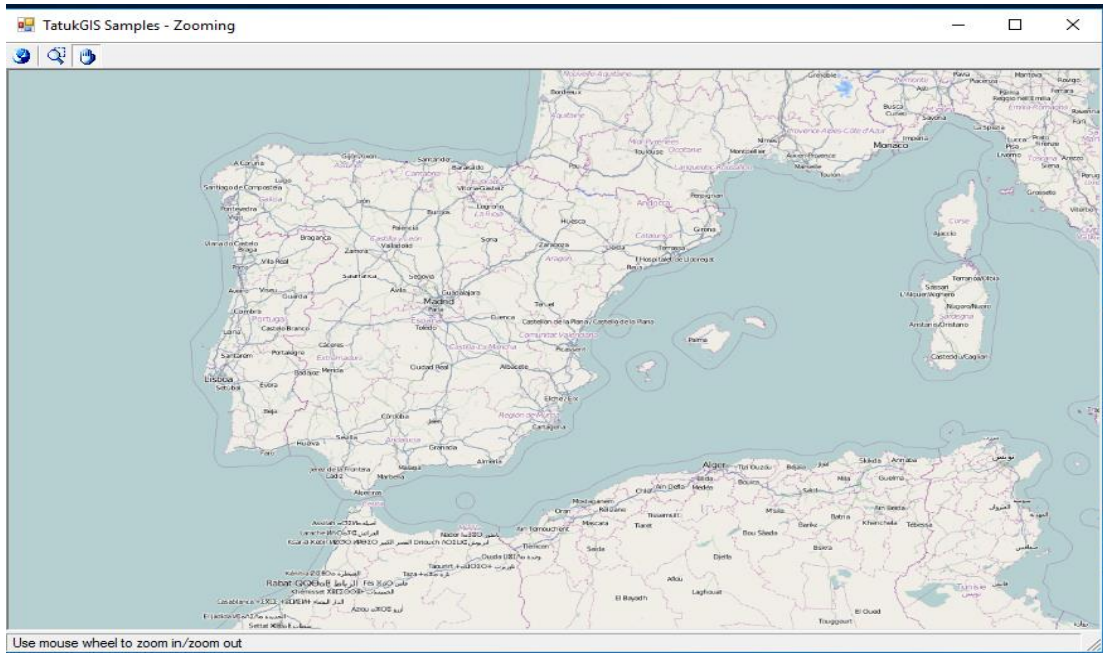


Figure 18: Tatuk OSM tiles in windows form

3.6 WPF (Windows Presentation Foundation)

The main reason for implementing in the WPF is that our whole company project in the WPF, so my company supervisor advised me to do it in WPF. Windows Presentation Foundation gives a very nice graphical representation of weather data with different types of layers or it's very easy to change the layer from one layer to the other layer, but it takes lots of time to load or update the layer for example if user switch the layer from temperature to sea water velocity it takes lots of time to update the layer. WPF gets the layers from the online server and thus, it takes the time to update. OSM tiles and layers in the WPF is as shown below.

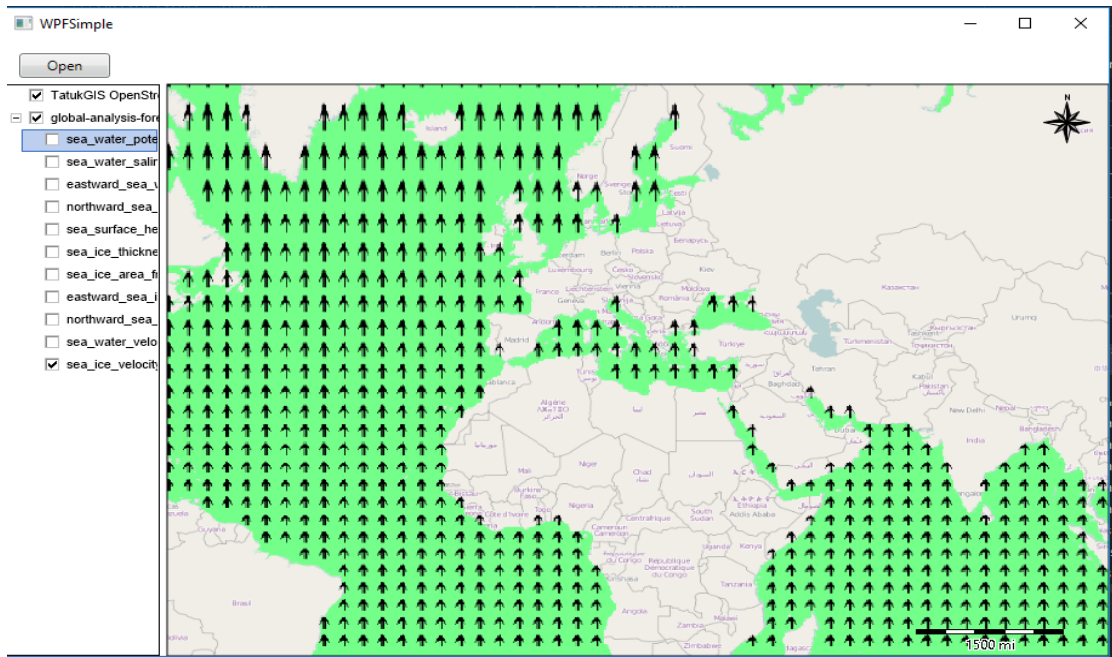


Figure 19: WPF with OSM and Layers

3.7 Sharp Map

Sharp Map is a simple to utilize mapping library for use in web and desktop applications. It gives access to many sorts of GIS information, empowers spatial questioning of that information, and renders lovely maps. The engine is composed in C# and based on the .Net 4.0 framework. [25] In the sharp map, the OSM (open street map) moves smoothly when compared to the other OSM GIS engines. Adding the WMS layers to the Sharp map is a really easy process, but its cost is too high.

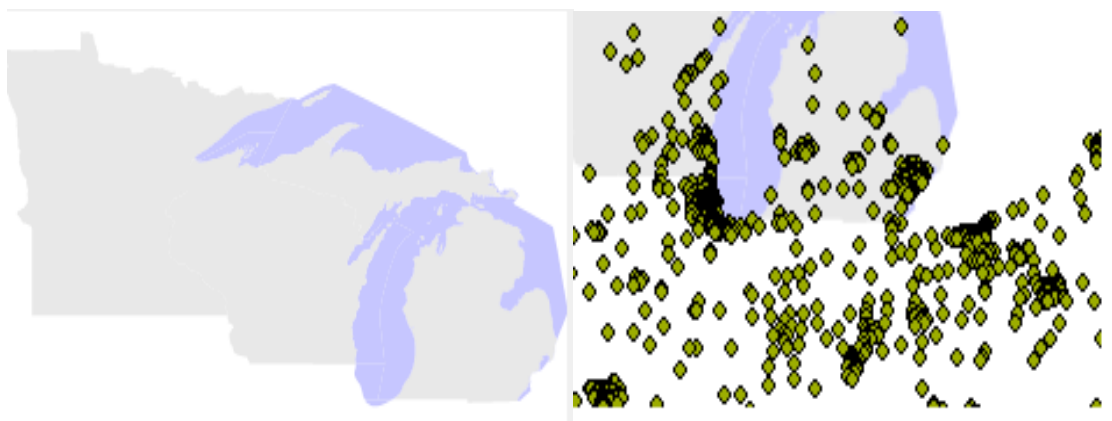


Figure 20: Sharp Map with WMS layers

3.8 Conclusion for Selecting GIS

In above sections, we saw two GIS engines that are Tatuk GIS and Sharp Map. In Tatuk GIS it's easy to apply the layer directly from the server. On the other hand, in the sharp map, it's easy to apply the layers from the server. However, the one of the main reason for selecting the Tatuk GIS is that sharp map is costlier than the Tatuk GIS and moreover later company decided to work with the Tatuk GIS, thus we continued with the Tatuk GIS in our project.

The following chapter continues with the marine data downloading the data from the Copernicus by using the python script.

Chapter 4: Methodology and Management

4.1 Overview

This chapter reviews various project management practices that were used during the earth observation life development process. Various project methodologies such as waterfall, agile and incremental were reviewed and evaluated on which was best choice for the project. During development, the project was divided into small manageable parts and organized into work breakdown structure for easy development.

4.2 Project Methodology

This chapter depicts the development methodology used in the development of the project, and its planning and management. There are many steps in the software development which include the requirement, analysis, design, and development. Deployment and maintenance phase. However, most of the time will be spent on analysis, design and development phases. However, the other phases such as requirement phase where requirement collection and elicitation are done are crucial as they form the basis of the next phases. Failure to do a thorough work in requirement phase leads to various revisions of the phase when the project has already started. Revisions of the previous phases are restricted by some software development life cycles. These models include the traditional linear models such as waterfall methodologies. However, other methodologies such as agile, incremental and prototyping process models allow iterations. Software lifecycle iteration help improves the final product and thus the client is well satisfied [26]. So as to fulfill the customer requirements, they should follow all phases in the development. Customer collaboration is allowed in agile methodologies.

Waterfall methodology also known as the traditional model is a linear sequential software development methodology that decomposes the project into various phases designed to perform a particular activity. Each phase of waterfall model must be completed before the successor phase is started. The model is usually used when the requirement is not going to change or there are not requirements uncertain. Each phase is reviewed independently so as to determine if the project is on the right track[27]. Also, the phase must be reviewed as once the next phase is started the previous phases are closed and can be revised. The model is typically applied when it is known at the outset that it is possible to carry out the complete survey of

the requirements of a project. This means that most of the system requirements, and the inputs and outputs that it has with external systems are well known (if any) [26]. The flow of the development of the project is as shown in the below waterfall model[28].

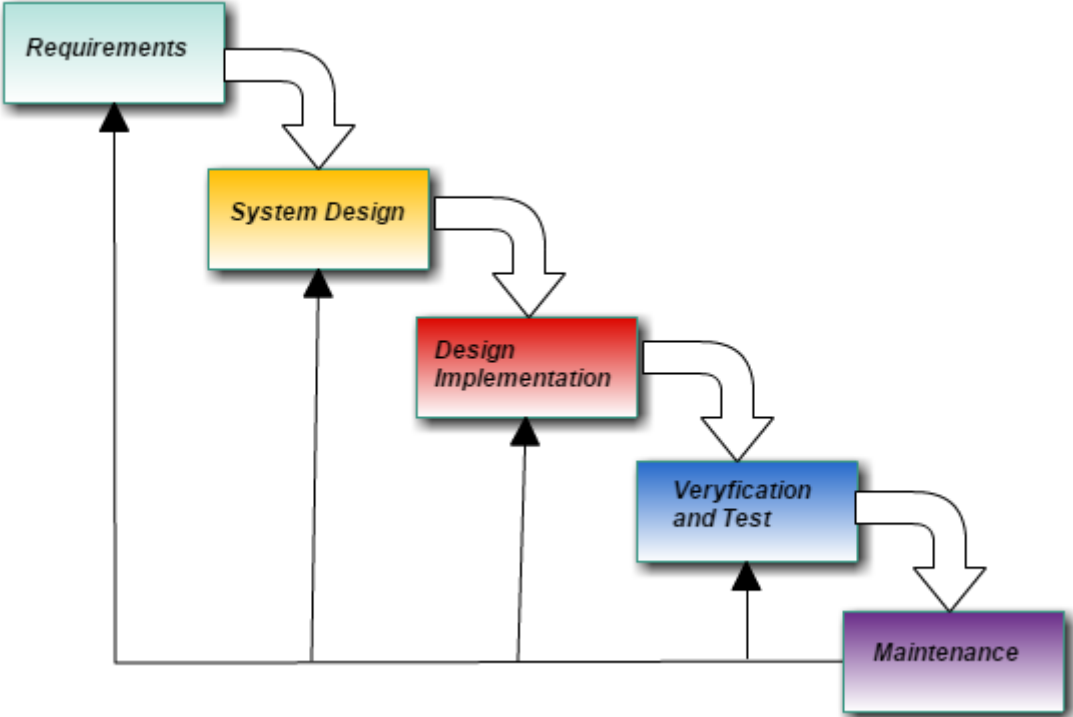


Figure 21: waterfall model of the software development model (SDLC)

Incremental software development life cycle involves incremental requirement analysis, designing and implementation and well as testing. These phases are added incrementally till the project is closed. When all the system increments as specified in the model have been attained, then the project is marked complete. The incremental model divides the entire project into small manageable project increment which is developed incrementally and given in form of releases [29]. The increment is released to the market for partial use and they wait for the main project which is finalized after several other increments. This way the product is allowed to enter the market earlier than when using traditional model. The model phases of specification, development and validation phase are interlinked as shown in the figure below:

The incremental model applies the use of multiple linear sequences phase just like waterfall model. The sequential phase improves the functionality of each system release until the entire planned functionality is achieved. The sequential phases of the incremental model are shown in the figure below[30].

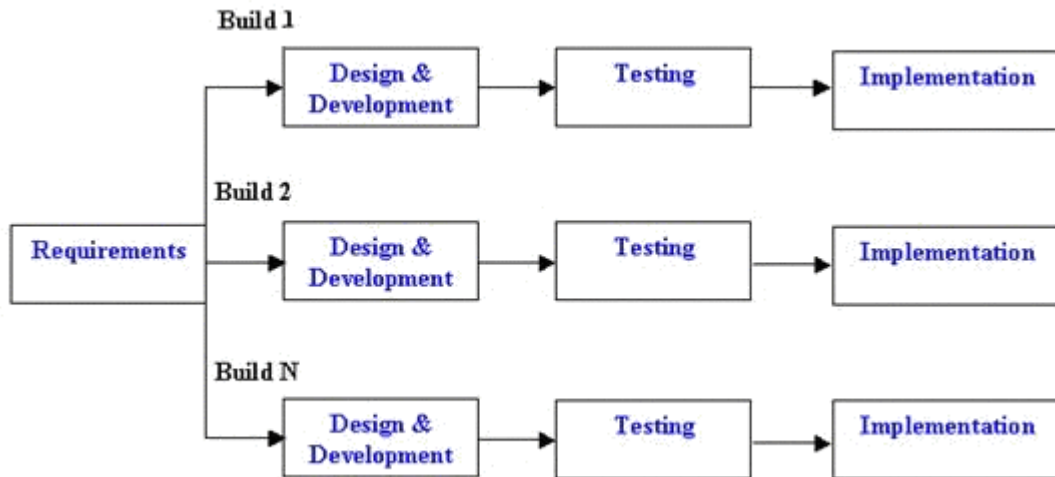


Figure 22: Incremental Model of the software development model (SDLC) showing various increments up to nth increment

Another widely used model known as agile model encompasses various process such as Scrum, Extreme programming, rapid application development, enterprise agile and agile unified process. The agile model involves a set of practice such as iterative practices, adaptive planning, teamwork, customer collaboration as well as early delivery. As a result, the process model gives a perfect product which has severally been reviewed by the client who is part of the development team and given feedback for improvement. Agile approach releases a small increment after every iteration which is given to the client for review[31]. The agile approach is non-linear as it forms a cyclic process that keeps repeating itself as a result of iterations as shown in the figure below [32]. Several cycles results to various releases and improvement of the previous release functionality leading to perfection. The previous releases are well tested with client-centered approach and feedback used to improve the quality of the product. Agile practices exhibit highly related dependencies which need a strong and technical development team. The skills will be used to ensure good business requirement analysis. The design and development phases of the model require highly skilled and experienced teams of designers and developers. However, such requirements of the development increase the cost. Also, the ability to support adaptive planning which helps redo the requirement, analysis, design and development phases increase the cost of the project. However, iterations help improve the quality of the product. According to Gupta, regular changes affect the project structure and thus degrade the structure leading to project scope creep[33].

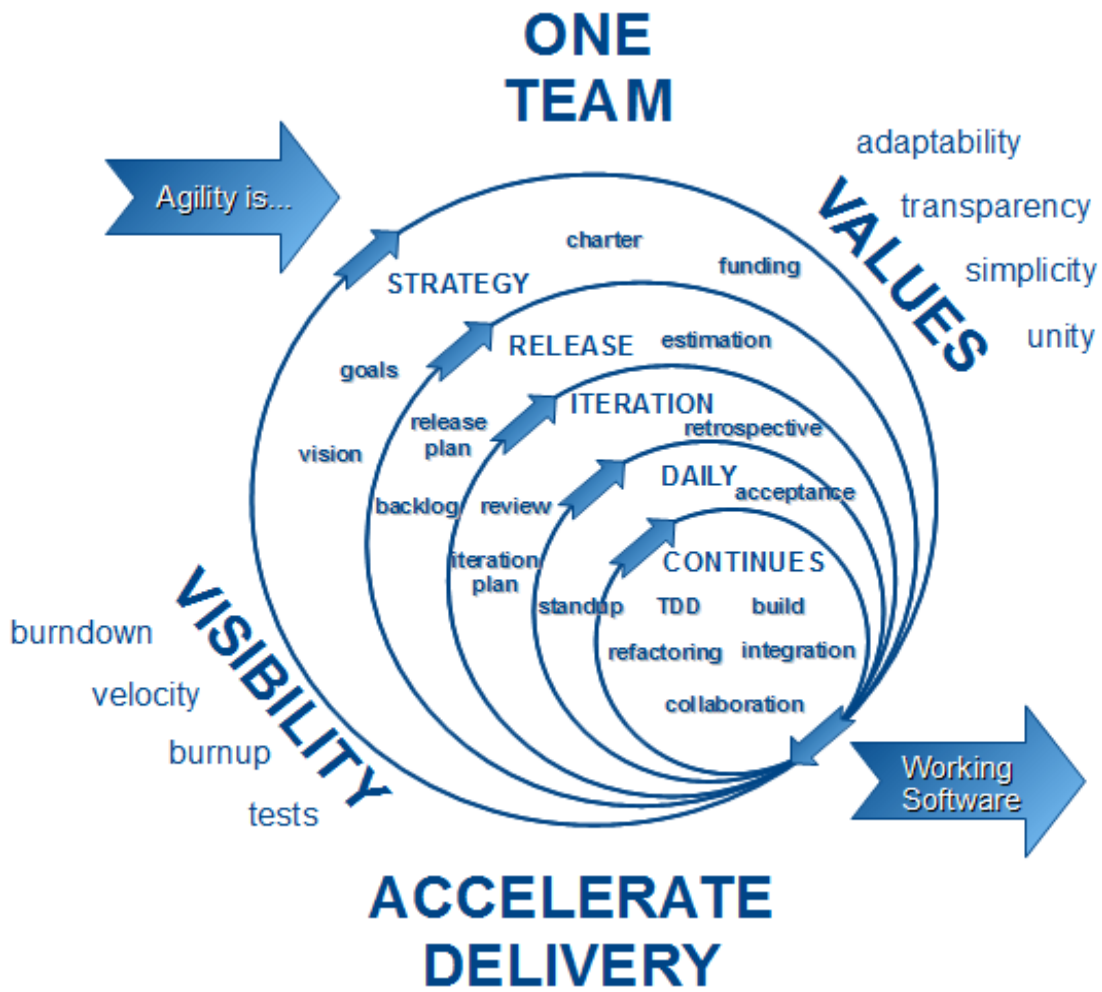


Figure 23: Agile Model of the software development model (SDLC)

Parameter	Process Model→	Waterfall Model	Incremental Model	Agile Model
Clear Requirement Specifications		Initial level	Initial level	Change incrementally
Feedback from user		No	No	No
Speed to change		Low	High	High
Predictability		Low	Low	High
Risk identification		At initial level	No	Yes
Practically implementation		No	Low	High
Loom		Systematic sequence	Iterative sequence	Highly customer satisfaction and incremental development[09]
Any variation done		Yes-v model	No	No
Understandability		Simple	Intermediate	Much complex
Precondition		Requirement clearly defined	Core product should clearly define	No
Usability		Basic	Medium	Most use now a days
Customer priority		Nil	Nil	High
Industry approach		Basic	Basic	High
Cost		Low	Low	Much Expensive
Resource organization		Yes	Yes	No
Elasticity		No	No	Very high

From the above methodologies, it is clear that agile approaches will be best suited for product development. The models use agile practices such as adaptive planning, iterations, and customer collaboration. The project has gone several iterations which have been revised to get an improved and quality product. The customer collaboration, in this case, my supervisor always took a customer look at the product from a customer perspective. The first product of the weather forecast applied layers on top of the base map using the Tatuk map. The Tatuk editor was used to merge base map with web map service (WMS) layers. The product graphical user interface allowed the user to click and view data. When a certain point illustrated by coordinates a checkbox was opened and layers were listed. This product was tested for the functionality and forwarded to my supervisor for further quality assurance testing.

The first application allowed the details to pop up with specific numeric information of the specific point clicked by the user. This representation was done in the windows form. However, my supervisor noted some user problems on navigating the background window when the pop-up window is on. As a result of the feedback, a second cycle was started to perfect the product. In second release annotation representation was used to solve this problem which allows the user to move the background map when the annotation is on. However, just like a stubborn client, this was not all. The supervisor asked for a more improved application that could allow the viewers to see the weather for the last 15 days. These days were to be represented in a graph. These features called for a third-round iteration which in the provided a perfect product which was absorbed by the company as result.

The product was developed through a series of activities which involved

Requirements collection and analysis:

In this starting phase, the potential requirement of the application is methodically analyzed, and note down the analyzed specification document, in the later stage this document serves as the basis for all future development work. In brief, the requirements document defines what the application should do, but not how it should do it.

System Design:

In this second phase largely covers the technical design requirements, for example, programming language, data layers, services, etc. The specification of the design will typically create outlines how exactly the business logic covered in the requirement will be technically implemented. The main goal of this phase is to "get across to a screen" the various ideas of how the project will work. In this way, it is possible to avoid possible problems and ensure that what was designed is in accordance with the intended. At this stage, a design of the system/application architecture and the interactions that it has with external sources is also conceived;

Implementation:

The design of the system is verified and the implementation begins according to the design, the real source code is finally written in this phase i.e. implementing all modules, business logic, and service integrations that were specified in the previous stages.

Testing:

The system testers included my supervisor who assisted me in make necessary recommendations. The supervisor assisted to systematically discover and report the issues within the application that need to be corrected. This phase is the crucial phase because it requires repeating the previous coding phase in order to fix the bugs.

Maintenance:

This is the final stage where the project is already in production, and it will receive the necessary maintenance, mainly correcting the errors that have not been identified in the previous phases, and required to keep it functional up to date.

There are minor differences in the numbers and descriptions of the steps used in agile approach, depending on the developer. Regardless, the concepts are all the same and encompass the broad scope of what it takes to start with an idea and develop a full-scale, live application. In the above model, it's possible to go back to the previous phases at any stage, for example, in case it is found that a requirement needs to be updated/modified or in the event that a function is not implemented as intended. Even though this project is doing not have concrete clients, so this role was done by the company's advisor. Here he is the person that the requirements were defined and approved, as well as, the verification of the implementation of the same, that is if they were in accordance with the intended. In practice, at each stage, the objectives achieved were regularly evaluated with the company's supervisor in order to receive feedback and, if necessary moved back to an earlier stage.

4.3 Project Management

The project is large with limited resources. To ensure quality end product has achieved the project was divided into several activities with subtasks for easy development.

The internship began on September 12, 2016, and ended on June 12, 2017. During the first two months of the internship, Study of the state of the art regarding GIS engines in WPF, taking into account the identified requirements namely, WMS feeds integration. This phase served to acquire knowledge about the various GIS technologies that exist, how they could be integrated into the project and the various concepts inherent to this type of technology, such as the concept of feature and layer, or the concept of Map tile.

The second phase started on November 7, 2016. This phase consisted of downloading Copernicus data i.e. weather forecast data. During this phase firstly downloading the forecast data into the local drive of the system for testing purpose, the downloaded data was inserted into the MySQL database, it inserted the data successfully. Then the company supervisor advises me to download the forecast data directly into the company Oracle database. At end of this phase, we reached one of the major milestones in the project i.e. downloading the weather forecast data into the Oracle database every midnight.

The third phase started in the mid of the December i.e. on 16, 2016. This phase was to write the design document describing the work to develop. This phase helps to collect more information about WMS layer applications similar to our project i.e. Earth Observation. This phase ended after the improvements that were suggested from the supervisor of XSealence in Lisbon were applied.

The fourth phase of the internship began on January 5, 2017, with the implementation of the requirements set for the Earth Observation, and continued until the delivery of this document. This phase began with the development of the proposed solution, the main goal was to apply the WMS forecast layers on the top of the Tatuk base map. The next task is to get the weather information of the user clicked point. During the development phase, there were several discussions and meetings with the company advisor in order to improve the user interaction component.

These project activities were used to develop project work breakdown structure to ensure each activity is planned for given enough time.

	WBS	Task Name	Duration	Start	Finish
1	1	Project Initiation	24 days	Mon 9/12/16	Thu 10/13/16
2	1.1	Evaluation and Recommendation of project idea	5 days	Mon 9/12/16	Fri 9/16/16
3	1.2	Project charter development	12 days	Mon 9/19/16	Tue 10/4/16
4	1.3	submission of project charter	4 days	Wed 10/5/16	Mon 10/10/16
5	1.4	reviwing of project charter	2 days	Tue 10/11/16	Wed 10/12/16
6	1.5	Project charter Approval	1 day	Thu 10/13/16	Thu 10/13/16
7	2	Project planning	40 days	Fri 10/14/16	Thu 12/8/16
8	2.1	create preliminary scope statement	14 days	Fri 10/14/16	Wed 11/2/16
9	2.2	Determine project Teams	12 days	Thu 11/3/16	Fri 11/18/16
10	2.3	project plan development	7 days	Mon 11/14/16	Tue 11/22/16
11	2.4	Submission of project plan	7 days	Wed 11/23/16	Thu 12/1/16
12	2.5	approval of project plan	2 days	Fri 12/2/16	Mon 12/5/16
13	3	Execution	95 days	Tue 12/6/16	Sun 4/16/17
14	3.1	project kick off meeting	1 day	Tue 12/6/16	Tue 12/6/16
15	3.2	user reguirement verification and validation	9 days	Wed 12/7/16	Mon 12/19/16
16	3.3	system design	18 days	Tue 12/20/16	Thu 1/12/17
17	3.4	hardware and software procurement	4 days	Fri 1/13/17	Wed 1/18/17
18	3.5	development system installation	49 days	Tue 1/19/16	Fri 3/25/16
19	3.6	testing phase	7 days	Mon 3/27/17	Tue 4/4/17
20	3.7	Install live system	4 days	Wed 4/5/17	Sun 4/9/17
21	3.8	training of users	4 days	Mon 4/10/17	Thu 4/13/17
22	3.9	Go live	2 days	Fri 4/14/17	Mon 4/17/17
23	4	Control	32 days	Mon 4/17/17	Tue 5/30/17
24	4.1	project management	12 days	Mon 4/17/17	Tue 5/2/17
25	4.2	project status meeting	1 day	Wed 5/3/17	Wed 5/3/17
26	4.3	risk management	12 days	Thu 5/4/17	Fri 5/19/17
27	4.4	project plan updating	7 days	Mon 5/22/17	Tue 5/30/17
28	5	Closeout	33 days	Wed 5/31/17	Fri 7/14/17
29	5.1	procurement audit	1 day	Wed 5/31/17	Wed 5/31/17
30	5.2	documentation of lessons learnt	21 days	Thu 6/1/17	Thu 6/29/17
31	5.3	updating of documentations	9 days	Fri 6/30/17	Wed 7/12/17
32	5.4	gain formal acceptance	1 day	Thu 7/13/17	Thu 7/13/17
33	5.5	submission	1 day	Fri 7/14/17	Fri 7/14/17

Figure 24: work breakdown structure activities was used to develop Gantt chart

The work breakdown structure activities were used to develop Gantt chart to show duration for each activity plotted using horizontal bars.



Figure 25: The project Gantt chart

4.4 Conclusion for Project Methodology and Management

The project was developed using agile methodologies which allowed the project to be divided into small modules which allowed parallel programming. Earth observation system was a huge system and the company divided it into various parts and allowed to various team members. The project took about 9 months. The WBS activities were used to help guide through a systematic development of the system. Next Chapter describes about the Copernicus Marine Environment Monitoring Services.

Chapter 5: Copernicus (Marine Environment Monitoring Services)

5.1 Overview

This chapter going to discuss the Copernicus (European Commission). The first section of the section will discuss the Copernicus programs and then components of the Copernicus Marine Environment.

5.2 Copernicus Marine Environment

For our project, data will be downloaded from the Copernicus Marine Environment Monitoring Services. For the Earth Observation, there are mainly it has two major space programmers they are GALILEO (satellite navigation system) and COPERNICUS [34].

The COPERNICUS started as Global Monitoring for Environment and Security (GMES), which was established as European program for earth observation and monitoring. The second space program of the Earth Observation i.e. Copernicus. The Copernicus space program covers the three components they are mentioned below:

- SPACE;
- INSITU;
- SERVICES.

5.2.1 The Copernicus Space Component

The Copernicus space component consists ESA's (European Space Agency) "Sentinels". The sentinels are primarily being developed for particular needs of the programs running be Copernicus. The programs are used in contributing missions that are established by national, international for European nations to offer a pool of data for Copernicus services. The agency is in charge for Copernicus space components that collect the data. European Organization for the Exploitation of Meteorological Satellites (EUMETSAT) offers Copernicus unparalleled operational remit experience for tried and tested capacity for meteorological satellite data, services and products[34].

5.2.2 The Copernicus InSitu component:

The Copernicus monitoring service which depends on data that is extracted from InSitu monitoring networks components such as ocean buoys, air quality monitoring network, maps, and ground-based weather stations. The monitoring network components offer powerful; integrated data and information for measuring and validating satellite data. For instance, moorings and surface measure a wide range of ocean sea surface options such as water salinity, water velocity, temperature, ice velocity, sea currents over a given period of time. The measured data is used by various bodies to make important decisions in their various day to day work life. InSitu networks are managed by European state members and international bodies who make the weather data available for forecasting services as agreed in their policies. Another agency named, European Environment Agency (EEA) is said to be main leading Copernicus FP7 GISC project Work for cataloging the requirements of the InSitu networks. The EEA agency offers Copernicus survives, pilot agreements as well as frameworks for accessing all relevant data in a sustainable and timely manner[34].

5.2.3 The Copernicus Service component

The third component of the Copernicus provides various services to help provide security, land monitoring, climate change, emergency management and atmosphere monitoring. The monitoring services are done the Space Component Sentinel which offers a set of Land Sea and atmospheric observations. Sentinel 1 was launched to offer high-resolution, all-weather, 24/7 and unique set of observation for both ocean land services[34]. Similarly, Sentinel 2 offers a high-resolution optical imagery for soil, water, and vegetation imagery. It also offers coastal areas and inland waterways imagery as well as emergency services. Sentinel 3 offers marine and land highly accurate radar and optical altimetry data such as sea surface topography, sea and land surface temperature. As a result of high accuracy, the data have high reliability. According to Copernicus Sentinel 4 will the third generation Meteosat to be managed by EUMETSAT for atmospheric monitoring of components [35]. Sentinel 5, which will be second generation Meteosat will be also be managed by EUMTSAT

Among these services land, monitoring and emergency management are already functional while atmosphere monitoring and marine monitoring are still in a pre-operational mode, or in a development phase. The climate change monitoring and services for security applications[36], Moreover, these services have reached different degrees of maturity.

5.3 Technologies used

Several different technologies and tools were used to develop the Earth Observation. The development platform used was .NET (version 4.5.1) because it was the most commonly used development environment in XSealence. In this way, it was possible to integrate with the existing project, the visual studio that allows receiving the information about the weather forecast from the database.

The IDE (Integrated Development Environment) that was used for application development was Visual Studio 2016[37], which owns Microsoft and which at the beginning of the implementation was the latest version of Visual Studio. Tatum 4.6.0.9975 editor was used for merging base map the with the weather forecast layer from the server.

The project backend which will be downloading the weather forecast data was developed using Python script, the Python version 2.7[38] is used for downloading the data from the Copernicus server, and also used motu-client-python-1.0.8-delivery for deploying. Motu is a highly efficient and robust Web Server which fills the gap between heterogeneous Data Providers to End Users. Motu handles, extracts and transforms oceanographic huge volumes of data without performance collapse[39].

5.4 Workflow Architecture of both Frontend and backend

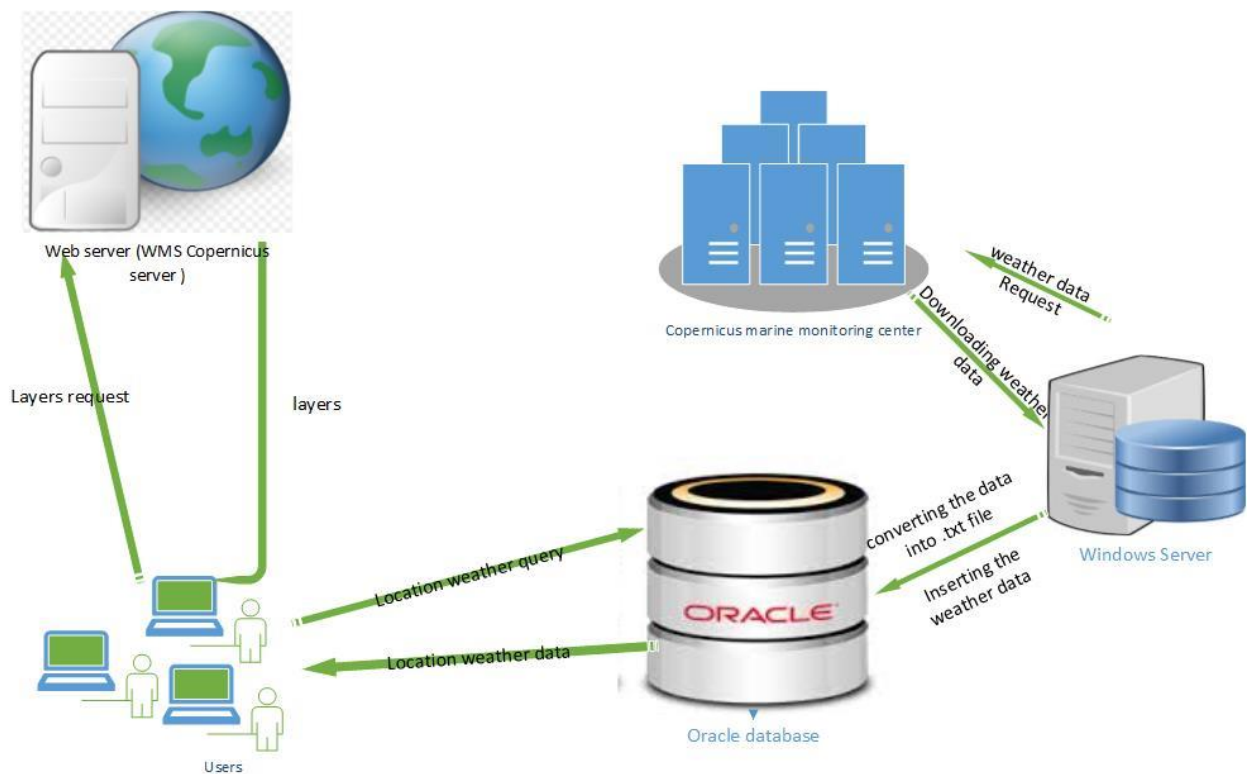


Figure 26: Earth observation desktop application workflow diagram

The figure above shows the complete overview architecture of the earth observation, in the above figure right-hand side is the backend and on the left-hand side is frontend part as discussed in the earlier section in the front-end part it's getting the layer from the server and fetching the weather data of the clicked point from the database, firstly the user will apply the layer and then fetch the weather forecast data from the database of the clicked point. In the left-hand side, it was done in the python script, the main function of the script is to download the weather forecast data into the database its everyday process, the script runs on the company server, when it runs the script its contact with the Copernicus dataset in order to download the weather forecast data in this way both front end and backend are linked together.

5.5 User case diagram

Use case diagram will show system element interaction. The use case diagrams helped in system analysis to identify, clarify and organize the system requirements. These diagrams show the behaviors of the system or system users. The diagram helps understands what users

should perform in collaboration with external users to offer some observable and valuable result[40].

Users:

- i. The user can login;
- ii. The user can view weather options;
- iii. The user can choose weather options;
- iv. The user can choose the particular coordinate;
- v. The user can get the information for particular coordinate and also previous 15 days info.

System:

- i. The system can show the map variables;
- ii. The system can show the map;
- iii. The system can show the weather layers on the map;
- iv. The system can Get the user clicked coordinates;
- v. The system can send the user clicked point to the database;
- vi. The system can get the data of user clicked point;
- vii. The system can generate the graph.

Server (python script):

The server can download the weather data. This is downloaded every midnight;

The server can insert the data into the Oracle database;

The server can convert the data of .NC file format to the text file while inserting.

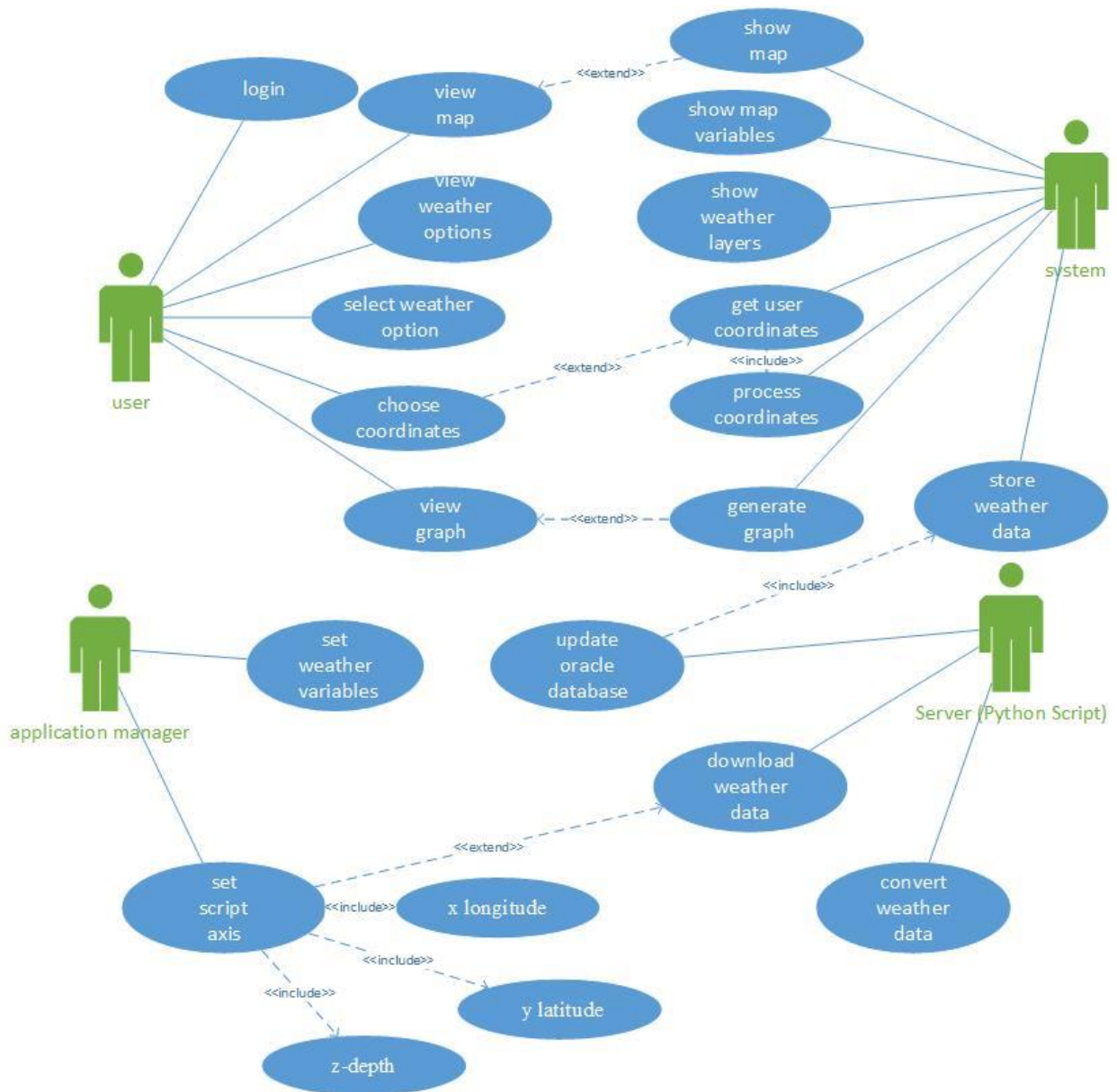


Figure 27: The system use case diagram

The application manager can set scripts which can include setting:

The application manager can set the Z-depth;

The application manager can set Y latitude;

The application manager can set X longitude;

The application manager can set weather variables.

5.6 Data Flow Diagram

Data flow diagram is a graphical representation of how data flows in the system. The diagram models all steps of system process aspects. The diagram illustrated how system inputs are processed to outputs. There are various levels of data flow diagrams such as level 0 DFD which is popularly known as context diagram, level 1 and level 2 DFD [41]. The following diagram is a level 1 DFD showing the detailed processes of checking a coordinate weather forecast.

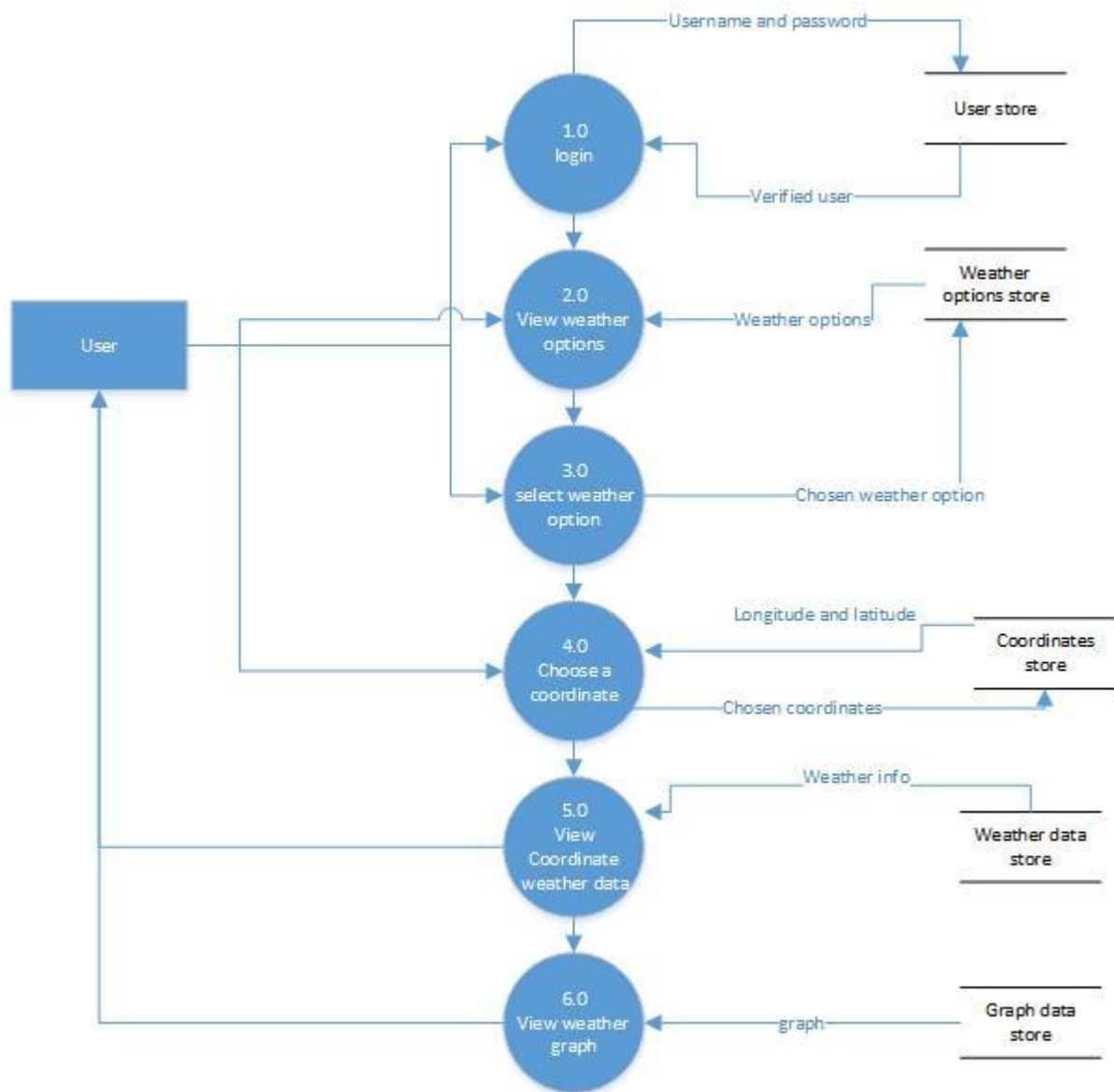


Figure 28: Earth Observation level 1 Data Flow Diagram

5.7 Class diagram

The application will be developed using Object-oriented Programming and thus coding will be build based on the object classes. Class diagrams to represent the application objects will describe the system structure by showing the application classes which are composed of class characteristics (attributes) and methods (class operations) [42]. The classes will be linked through relationship showing the how the objects associated.

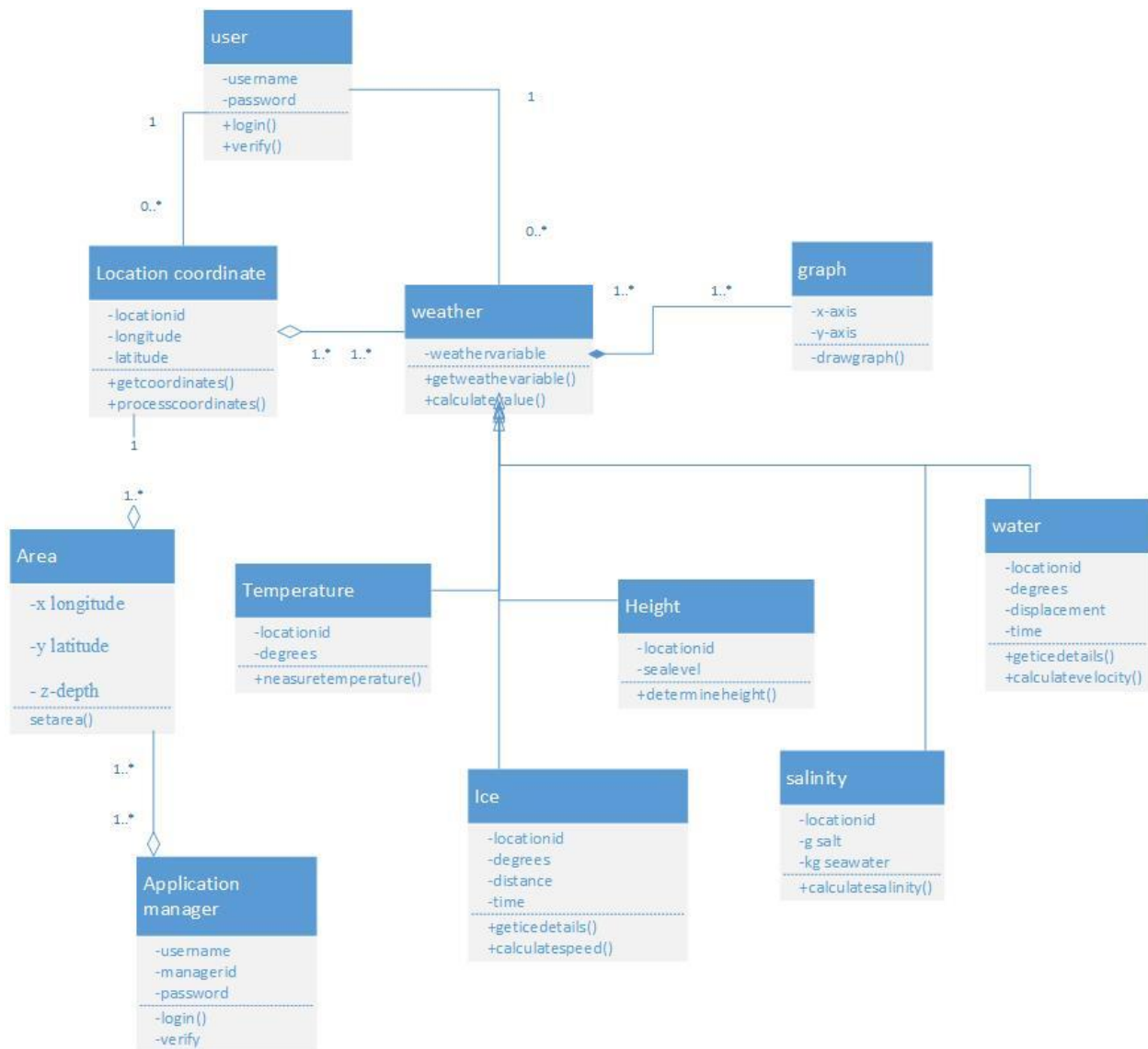


Figure 29: The desktop application class diagram

The class diagram above has various classes and their relationship shown. There are various relationships that may exist between classes. They include inheritance, composition, aggregation, association, and realization.

Class association exists between user classes, location coordinate classes. The relations are considered the basic relation which connects the classes in any way. Also, association relation exists between user and weather data class [43]. A user may request for weather data. The request is processed through the weather data class which acts as the parent class for salinity, temperature, height, ice and water classes. Salinity, temperature, height, water objects belong to respective class names. This means these classes considered as child or subclasses inherit the attributes and methods of the parent class weather data. The data fetched by the classes are used to plot a 10-day graph. This means if weather class does not exist to provide weather

data, then the graph class do not exist [44]. This means weather and graph class are related through association. If the class weather is destroyed then, graph class is also destroyed. Weather class and location coordinates are related through aggregation relationship. This means the two classes are built as a collection, just like composite relations. However, in aggregation relation, the contained class can exist without the container class, unlike in composition class, in which destruction of container automatically leads to the automatic destruction of contained class [45].

Other classes such as Salinity, temperature, height, water is related to location class through association as this class information is retrieved in relation to a specified location.

5.8 Conclusion for System Design and Copernicus

In the above sections discussed about the system design that determines how well the system will be implemented. Poorly designed system result to systems that do not meet the customer expectations. System analysis and design such as use of use case diagrams and data flow diagrams guide both the developer and client to know how data will be processed within the system. System architecture diagrams was also designed to illustrate how the system downloads data from Copernicus web map services in .NC form and converts it in .txt form which is storable in the oracle databases. This data is queried during marine weather forecasting. This an object-oriented programming project, class diagrams were identified and design for easy implementation of the front-end part and application. Next chapter going to deal with the back-end development of the project.

Chapter 6: Backend Development

6.1 Overview

This chapter discussed the backend part of the project. The application back-end was tasked with downloading of the weather forecast data every midnight. The downloaded data will be fed into the company's Oracle database. The chapter is organized into two sections. The first section discusses the data download from the Copernicus then the second section discusses by inserting the data into the database. The data is downloaded in files of .py form which cannot be directly inserted into Oracle database [46]. As a result, the files are converted before inserting into .txt files by the web server and then stored to by the Oracle database.

6.2 Downloading data from Copernicus

As we discussed in the above section about the Copernicus, has many kinds of weather and climate monitoring products and services for the marine environment[36]. In the previous chapter is discussed about the WMS layers. The layers get data from the Copernicus web map services Earth Observation desktop application project, downloads this numeric data of the particular layer and stores into our application database.

From the Copernicus service, mainly we are downloading the data of this product, all information of the product is listed below:

- `product_id = "GLOBAL_ANALYSIS_FORECAST_PHY_001_024";`
- `dataset_identifier = "global-analysis-forecast-phy-001-024";`
- `motu_sc="-m http://nrtcmems.mercator-ocean.fr/mis-gateway-servlet/Motu";`
- `serv_id="-s http://purl.org/myocean/ontology/service/database#+product_id+"-TDS";`
- `dataset_id="-d "+dataset_identifier.`

In the above-mentioned product, there are two services available, the first service provides the data once in a day and in the other services it provides the data for every one hour[13]. This product includes daily mean files of temperature, salinity, currents, sea level, mixed layer depth and ice parameters from the top to the bottom over the global ocean. It also includes hourly mean surface fields for sea level height, temperature, and currents.

The second service product i.e. updating the data once in a day, which is having more than 10 variables. The system will download a variable like which is most helpful for the marine monitoring. The list below shows all the available marine weather variables to be contained in the downloaded product:

- sea_surface_height_above_geoid;
- northward_sea_water_velocity;
- ocean_mixed_layer_thickness_defined_by_sigma_theta;
- sea_water_potential_temperature;
- eastward_sea_water_velocity;
- sea_ice_thickness;
- sea_ice_area_fraction;
- sea_water_salinity;
- eastward_sea_ice_velocity;
- northward_sea_ice_velocity.

6.3 Motu Client

Motu is a highly efficient and powerful Web Server which fills the gap between heterogeneous Data Providers to End Users. Motu client mainly handles, extracts and transforms oceanographic huge volumes of data without collapsing the performance. It enables to extract and download data through a python command line[39]. The forecast data file is not downloaded directly into the drive, but it is deployed in the Multi-client Server. To download data into our drive, download the Motuclient then give the path for the Motuclient in the python script as given below.

Replace {path_to_your} with your current path to motu-client.py = where you deployed it

- motu_cl="C:\abc\abc\motu-client-python-1.0.8-delivery\motu-client-python\motu-client.py -q".

6.4 Implementation

Downloading of the weather data in the application backend part play one of the major roles in our project. The data that is downloaded is in the numeric form of data. Numeric data is related to the frontend layers that is applied on the top TatukGIS map. The Python script is used to download the numeric data from the Copernicus in the backend.

The downloaded data from the Copernicus is in the .NC file format which is shown in figure 30. The data in the .nc form is not possible to be inserted in Oracle database with its present

form. As a result, this kind of format into the Oracle database is converted to .txt form contained in a text file format by using statement “from netCDF4 import Dataset, num2date”.

Name	Date modified	Type	Size
CMEMS_PSY4_Daily_Forecast_for_production_2016-12-20_forecast_20161220_bottomT.nc	20-Dec-16 12:41 PM	NC File	15 KB
CMEMS_PSY4_Daily_Forecast_for_production_2016-12-20_forecast_20161220_so.nc	20-Dec-16 12:40 PM	NC File	16 KB
CMEMS_PSY4_Daily_Forecast_for_production_2016-12-20_forecast_20161220_thetao.nc	20-Dec-16 12:40 PM	NC File	16 KB
CMEMS_PSY4_Daily_Forecast_for_production_2016-12-20_forecast_20161220_vo.nc	20-Dec-16 12:41 PM	NC File	16 KB
CMEMS_PSY4_Daily_Forecast_for_production_2016-12-20_forecast_20161220_zos.nc	20-Dec-16 12:40 PM	NC File	15 KB
CMEMS_PSY4_Daily_Forecast_for_production_2016-12-23_forecast_20161223_thetao.nc	23-Dec-16 3:31 PM	NC File	16 KB

Figure 30: Weather data in .NC file

Downloading the whole world forecast data into the database usually its take too long time, so downloading the forecast of some particular areas increases the speed of download. For example, the **application managers** need to set the axis in the script it’s as shown below.

Area: x longitude, y latitude, z-depth:

- xmin="-x -13" xmax="-X -7"
- ymin="-y 36" ymax="-Y 43"
- zmin="-z 0.494" zmax="-Z 0.4942"

Setting the particular area will boost the speed of downloading data as set variables like temperature, salinity, ice speed etc. as well. For this reason, the most important variables which help in marine or fish monitoring services are needed. Such variables need to be set as they are the only variables which are required are useful. Setting all marine weather data variables is of no use, hence, setting the particular variables in the Python script is as shown below.

Variable: -v toto

- var_cmd=["-v thetao", "-v zos", "-v so", "-v bottomT", "-v vo", "-v vsi"]

Where:

Thetao: sea_water_potential_temperature, zos: sea_surface_height_above_geoid;

So: sea_water_salinity, bottomT: sea_water_potential_temperature_at_sea_floor;

VO: northward_sea_water_velocity, vsi: sea_ice_velocity.

Instead downloading all variables data, the above idea is used. It also helps to increase the speed of download and also save the space in the database instead storing all variable information with some information being less relevant.

Since we are downloading the weather data every day, thus its requires to verify whether the downloaded is up to date or not. To verify if dataset is up to date as shown below.

```
dayj0=dt.datetime.today().strftime("%Y-%m-%d")
```

```
daylf=(dt.datetime.today()+dt.timedelta(days=forecast_range)).strftime("%Y-%m-%d")
```

```
date_cmd='t "%s 12:00:00" -T "%s 12:00:00"'%(daylf,daylf).
```

The final name of the file which specifies in which manner we want to save with the specified name and splitter is as shown below.

```
ficout=pre_out+daylf.replace('-',")+ "_" +var_cmd[1].split()[1]+".nc"
```

6.5 Oracle Database

The Oracle DB is a relational database management system Oracle Corporation. The Oracle database is originally developed in 1997 by Lawrence Ellison and with the help of other developers. In the present era, Oracle is one of the trusted relational databases and it is one of the widely-used database engines.

The Oracle database system is built around a relational database framework in which data objects may be directly accessed by users (or an application front end) through structured query language (SQL). It has its own network component to allow communications across networks. Oracle database is also known as Oracle RDBMS and, sometimes, just Oracle. It has a fully scalable relational database architecture and is often used by global enterprises, which manage and process data across wide and local area networks[47].

The Oracle database runs on most of the major platforms which are listed below, and it has different software versions are available, based on the requirements and budget.

- Windows;
- UNIX;
- Linux;
- Mac OS.

Oracle database has the different editions, the developer can select the database according to their needs the Oracle database editions are hierarchically broken down as follows:

- Enterprise Edition: which can offer all features, including superior performance and security, and is the most robust;
- Standard Edition: this edition contains base functionality for users that do not require Enterprise Edition's robust package;
- Express Edition (XE): this is the lightweight, free and limited Windows and Linux edition;
- Oracle Lite: it's for mobile devices.

Oracle has had a customer oriented approach during development and release of Oracle9i. This helped them understand what clients want more, and as a result, a desired increment of the Oracle9i with default configurations and hardened database environment was released. The release also came with user manuals on how to securely configure the database. The manuals and accompanying documentation also offer post-installation locking and default account expiry information which was found to be very critical in database maintenance.

According to a survey conducted by Ponemon Institute, 80% of the interviewed companies said that their applications have become more susceptible to attacks [48]. Of these respondents, 54% agreed that the most common experienced attack in their enterprises in the last 12 months has been SQL injection, while the rest of attacks such as cross site scripting and cross site request forgery combined made 46%. This means that of all threats to Oracle database SQL injection is the most potent attack that our database should focus on avoiding and prepare for mitigation in the event the worst happens[49]. The figure below shows the likelihood of each attack according to the survey[48].

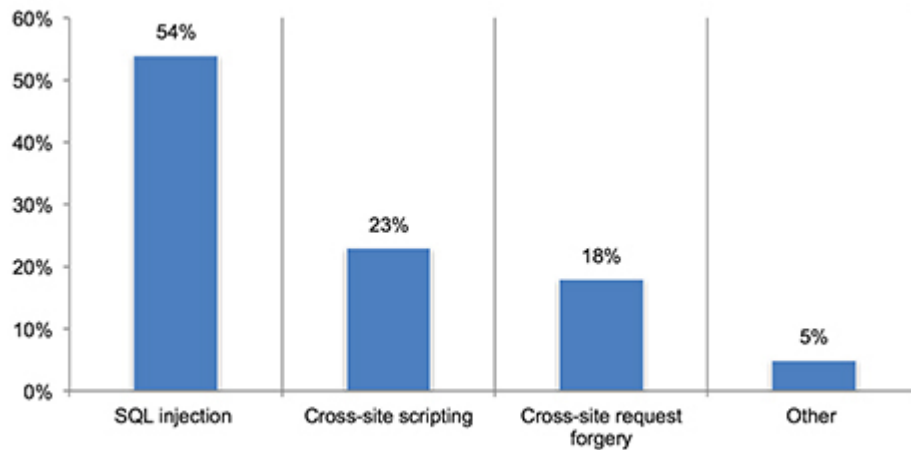


Figure 31: A bar graph showing how common is SQL injection is performed by attackers

From the figure, it's clear to conclude that SQL injection is the top hacking techniques that attackers have used to infiltrate our systems and steal sensitive and private data [50]. To avoid such SQL injection attacks, database systems need a detecting and blocking procedure which is possible as a result of high SQL grammar analysis performance engine Oracle database possess. The database has the capacity to determine if an incoming query statement is valid and if not, it is blocked.

Oracle database has an active, real-time firewall to provide blacklist, exception and whitelist policies as well as accurate and intelligent alarms that allow effective database monitoring with low costs. The firewall is as shown in the figure below [51]. The firewall form an independent database configuration and operation boundary to protect the database so as to reduce the risk of losing data and manage the changing regulatory landscape.



Figure 32: Oracle database firewall showing its alert, allow and blocking boundary

6.6 Architecture of Oracle DB

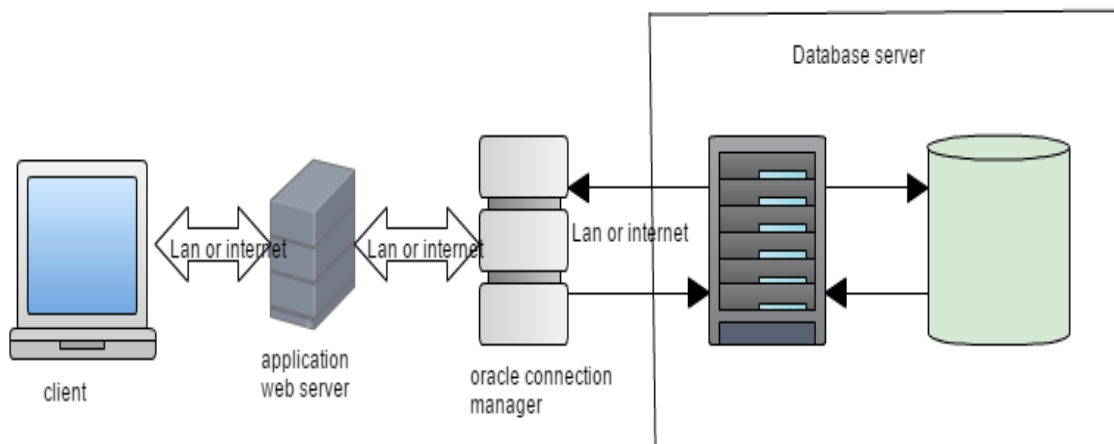


Figure 33: Architecture of Oracle DB

In Figure 33 shows the connection to an Oracle database server are made more scalable with Oracle Connection Manager and a shared server architecture. Oracle Connection Manager is used to offload some of the network I/O of the application Web servers, and a shared server is used to serve more concurrent users. The architecture of the Oracle database is split between the logical and the physical. The architecture which forms the large-scale distributed computing, also known as grid computing and the data location is irrelevant and transparent to the user, allowing for a more modular physical structure that can be added to and altered without affecting the activity of the database, its data or users. Sharing of resources allow

flexible data networks with the adjustable capacity to be calibrated down and up to fit demand, without reducing service quality[47].

6.7 Architecture for Downloading the Data

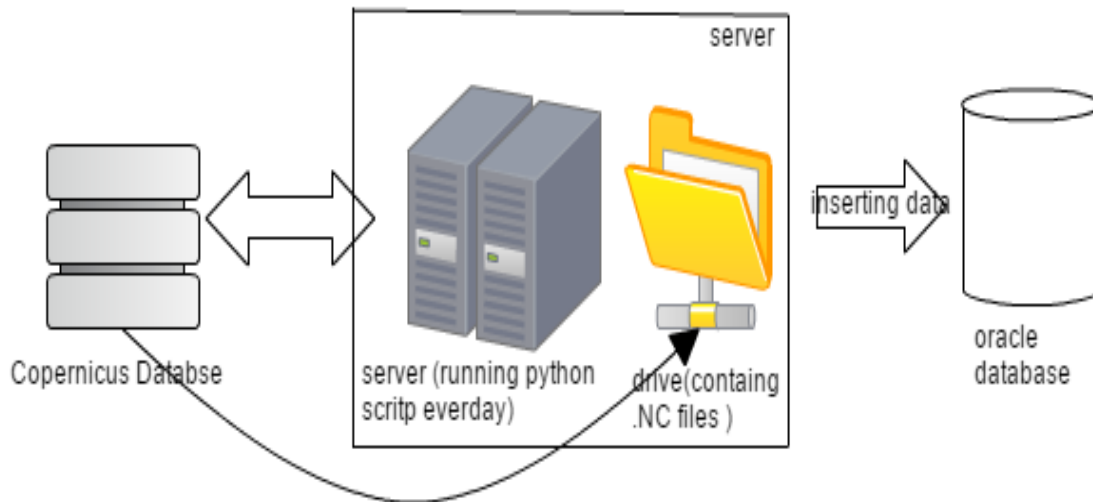


Figure 34: Downloading data Architecture

In figure 34, there are three components i.e. Copernicus database, server and Oracle database. As indicated earlier the back end part of downloading weather data from Copernicus was developed using Python scripting. The script is scheduled to run every midnight on the server. The script runs by contacting the Copernicus database with a particular product id, once the dataset is available then its download the forecast data file in the server local drive. While data is being downloaded it cannot be inserted directly into the Oracle database, because the Copernicus marine monitoring service provides the forecast data in the .NC file format. The .nc format files are not possible to insert the data into the Oracle database. To insert the data first we need to convert the .NC file format into the text file format, so once the forecast file is downloaded it's converted into the text format then it is inserted into the Oracle database.

6.8 Implementation

As per the above discussion, the Copernicus Marine Monitoring services provides the weather forecast data in the .NC file format, so once the file downloads its inserted into the database by converting the .nc format into text form. To convert the .nc files to file format the python script imports the dataset from the netCDF4 as shown below.

- from netCDF4 import Dataset, num2date

While Downloading the forecast data from the Copernicus the script inserts the data into the Oracle database. The process starts by connecting to the Oracle database. The connection to the Oracle database in the python script is as shown below,

- con = cx_Oracle.connect(ABC Admin/abc@abc.pt/ABC)

After connection to the database, the database is checked whether the database table IS existed or not. If the table already exists then its insert the data into that table. Suppose the table does not exist in the database first it's going to create a new table in the database then its insert the data into that created table. The database admin can set the number of lines to be inserted into the database as shown below.

- cursor.execute("CREATE TABLE FOREIGN.earth_observation_Data(DateTime date NOT NULL,longitude number(38,10) NOT NULL, latitude number(38,10) NOT NULL,type number NOT NULL, value number(38,10) NOT NULL) ").

Once the earth observation table creates, the columns look like as shown below.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	DATETIME	DATE	No	(null)	1	(null)
2	LONGITUDE	NUMBER(38,10)	No	(null)	2	(null)
3	LATITUDE	NUMBER(38,10)	No	(null)	3	(null)
4	TYPE	NUMBER	No	(null)	4	(null)
5	VALUE	NUMBER(38,10)	No	(null)	5	(null)

Figure 35: Columns in Database Table

In the above fig 53 COLUMNS_NAME refers to name of the items, DATA_TYPE refers to type of data to be inserted, COLUMN_ID refers to variable information for example in the above table 1 is for temperature, 2 is for surface height, 3 is for salinity, 4 is for temp at sea floor, 5 is for water velocity and 6 is for ice velocity, by using this column_id we can easily fetch the information the of that variable to the front end application. The column_id will be used as the primary key to uniquely identify each field.

Once the table is created then inserting the data into the database table as given below:

- `cursor.execute(`
 `"INSERT INTO FOREIGN.earth_observation_Datavalues(to_date('%s','YYYY-MM-`
 `DD hh24:mi'),'%f', '%f','%d','%f')"% (dateTime, lat, lon, var+1,`
 `data[lineToBeStored])).`

The inserted forecast data have to fetch it for the frontend of the application. To for fetch the data from the database to the front end of the application, we need to create the stored procedure in the Oracle database. The stored procedure in the Oracle is designed in such a way that it can search nearest latitude and longitude in the database with respective mouse click point on the map, then it will give value for that point which is near to mouse click point[52], so to find the nearest points in a database to a given point.

```
* rad2deg * (ACOS(COS(deg2rad * (p.latpoint))
* COS(deg2rad * (z.latitude))
* COS(deg2rad * (p.longpoint - z.longitude))
+ SIN(deg2rad * (p.latpoint))
* SIN(deg2rad * (z.latitude)))) AS distance
```

For example, let's consider the point with latitude 36.658 and longitude -23.59. The above oracle query finds the one nearest point to the given point in the order of distance.

6.9 Scheduled script on the server

The main function of the Python script is to download the current forecast data from the Copernicus servers. Since we need the updated weather forecast data every day a scheduled script download was developed to support this. Therefore for this task, the python script is ran in the Company server. The scheduler in the Windows server which runs the python script every day at 3 is in the early morning. The scheduled event in the server is as shown below.

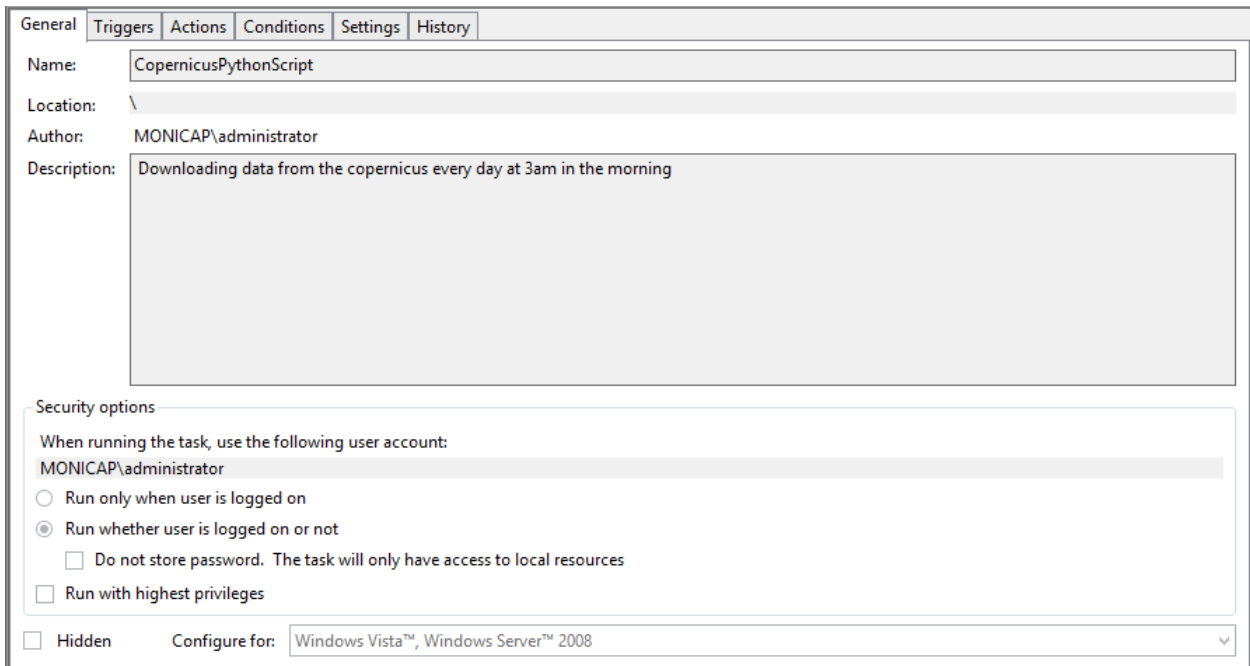


Figure 36: Task Scheduler in the windows server

In the figure below we are seeing two triggers because if it doesn't download properly or any interrupt to run the script on the server at 3 am, its cross check once again at 6 am whether it's downloaded properly or not. For instance, if it's not downloaded correctly at 3 pm then it's going to download correctly at 6 pm.

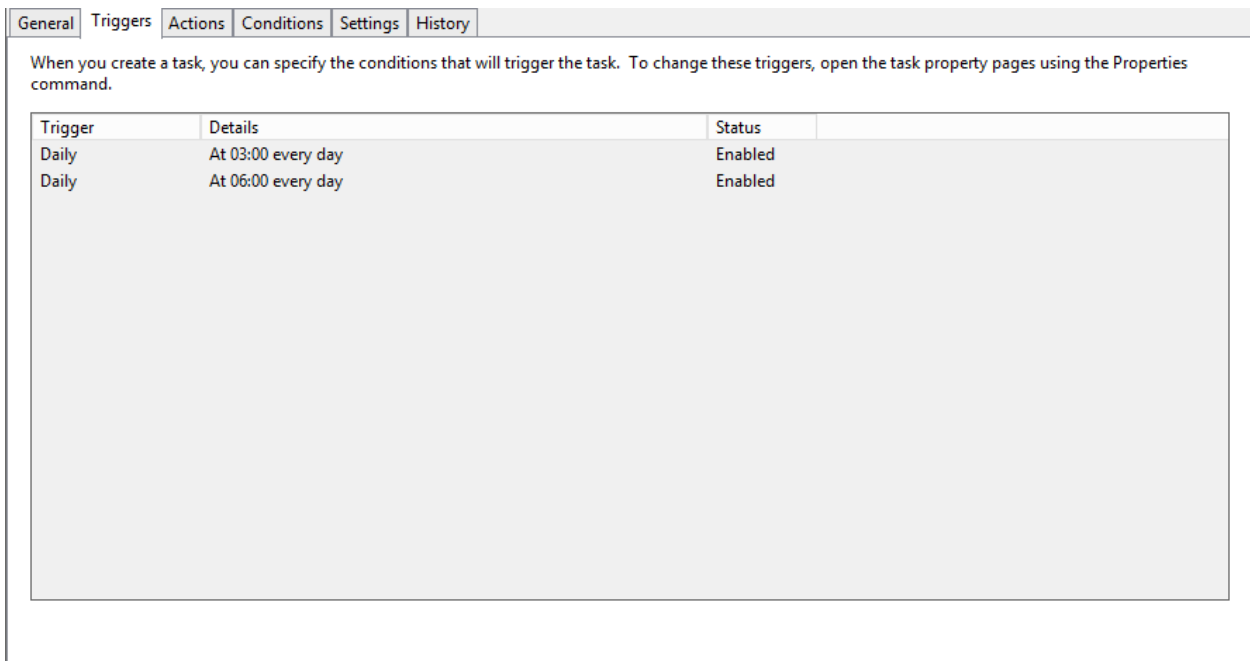


Figure 37: Multiple Triggers

6.10 Conclusion for Back-End Development

In above section discussed about the back-end development that the data queried by earth observation application comes from secure oracle database that was used to develop the system back end. The data in the database is downloaded from Copernicus database which collects daily weather data details. The data from Copernicus website cannot be held in oracle databases. The data is downloaded in .NC form which is converted into .txt form by python script and stored in the oracle database. In the next chapter is going to discuss about the front-end application. Next chapter deals with the front-end implementation.

Chapter 7: Frontend Development

7.1 Overview

In this chapter going to discuss information about the architecture of the WMS layers, information about the layers. The chapter also discusses how to apply that layers on the top of the Tatum GIS and as well as disabling or enabling. At the end, the chapter will explain the graphical representation of the forecast numeric data.

7.2 The architecture of WMS Layers

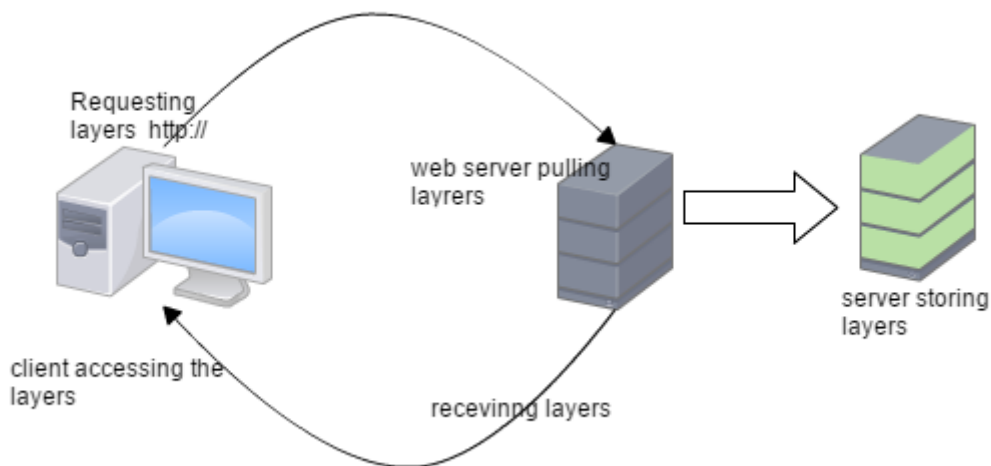


Figure 38: WMS Architecture of fetching layers

On the top of the map, the layer object communicates with the server to display the layers on the top of the map which is stored on the web server. In the above fig 38, the user requesting the layers from the web server as said earlier. The layers are stored on the server. After processing the user request, the web server sends the requested layers to the users. When the users zoom in or zoom out the new tiles of layers will be displayed on the top of the map, for this server will generate new layers corresponding to the zoom level. The WMS is the standard process of accessing layer data from the server where the layer data are stored. It generates layer data in the form of the image[11].

7.3 Applying Layers

The applying the weather forecast layers on top of the base map is very helpful for the users to view the weather condition of the particular area. Application of the weather forecast layers helps to take an overview mainly for the fishing activities. As discussed previously application of the top of the open street map, is done in the Tatuk Editor for testing purpose. The users can apply the layers of forecast on this project. This project is having the Tatuk GIS as the base map, above this base map user can apply the weather forecast layers.

7.4 About the Layers

These forecasts layers are provided by Copernicus Marine services. Copernicus Marine services also provides various services as shown below

CSW	C SW GetRecordById
WMS	global-analysis-forecast-phy-001-024 global-analysis-forecast-phy-001-024-hourly-t-u-v-ssh
SUBSETTER	global-analysis-forecast-phy-001-024-hourly-t-u-v-ssh global-analysis-forecast-phy-001-024
DIRECT GET FILE	N/A
FTP	N/A

Figure 39: Copernicus Services

In the above services, we are using WMS services in our project, the WMS services are as given below.

- Path=<http://nrtcmems.mercator-ocean.fr/threads/was/global-analysis-forecast-phy-001-024?SERVICE=WMS&VERSION=1.1.1&REQUEST=GetCapabilities>.

For applying the layers on the top of the base map, firstly took the tatuk base map in .ttkqp format.

Here .ttkqp is the tatuk extension. Using tatuk editor the base map was merged with WMS services layers. The merging of the map and services created one file of .ttkqp extensions that which includes both base map and forecasts layers. In the front end of WPF application, enable or disable options of the application were developed for the user. When the user clicks the button the window will open with checkbox and layers name list as shown in the below figure.

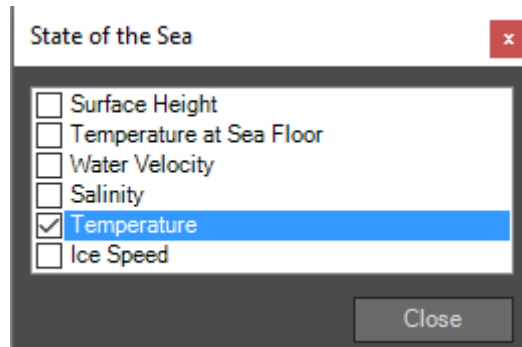


Figure 40: State of Sea window

When the user unchecks all the checkbox then the base map becomes active. This means only base map will visible.

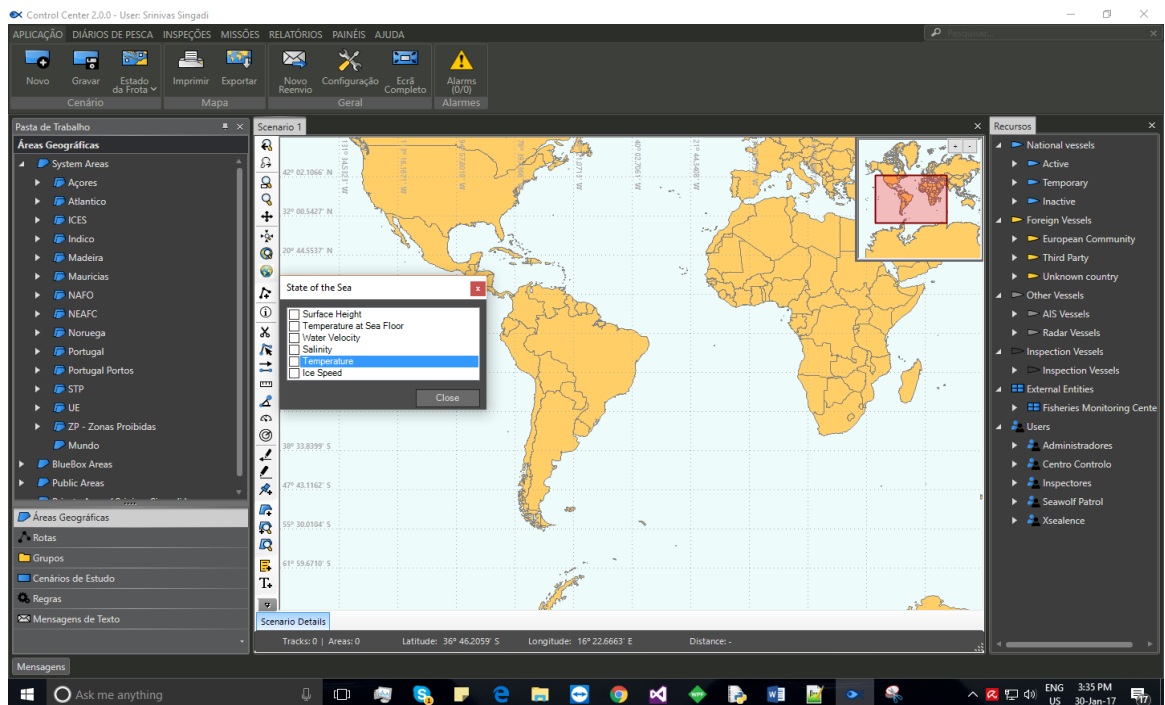


Figure 41: Base map (all checkbox unchecked).

When the user checks the checkbox with respective layers that particular layer will apply on the top of the base map, if the user selects another layer then previously applied will automatically be disabled (inactive) in order to prevent the overlapping of the layers.

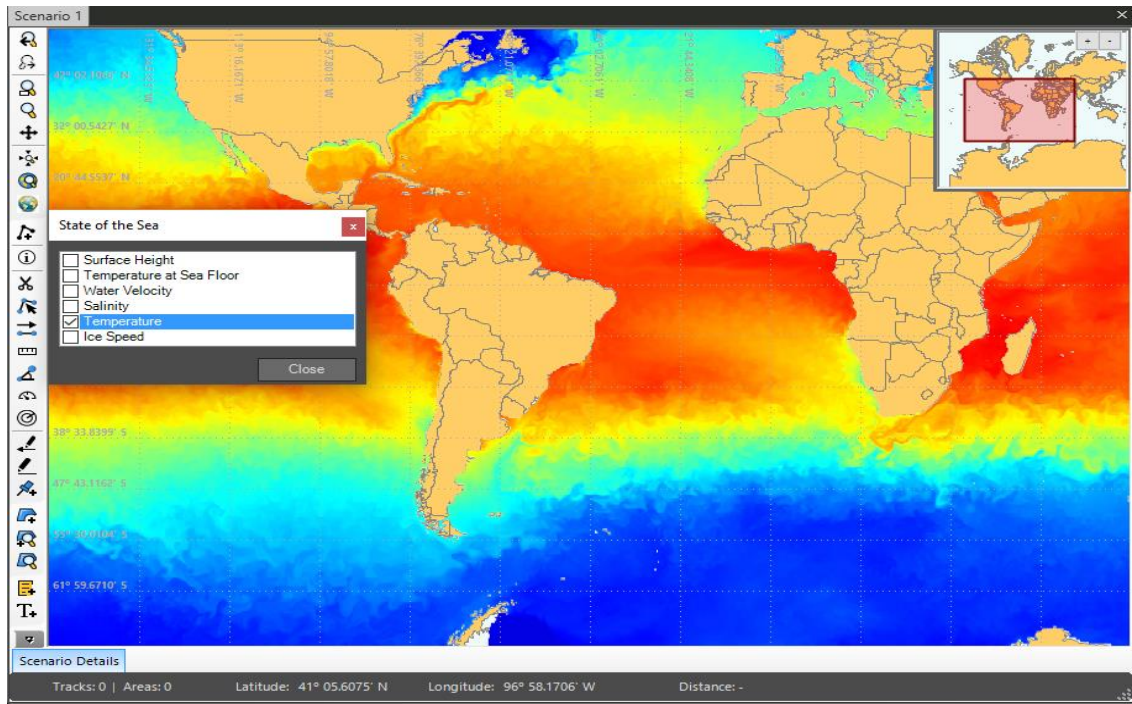


Figure 42: Base with Layer temperature

The code behind this layer is that, when the user clicks the check box of the layer, it gets into the forecast layers of the WMS services one by one to apply the exact layer on the top of the base map, the code is as given below:

If

```
(this.GetActiveSubLayerNames("global-analysis-forecast-phy-001-024?SERVICE=WMS&VERSION=1.1").Count > 0)
```

```
{
```

```
String activeSubLayerName = this.GetActiveSubLayerNames("global-analysis-forecast-phy-001-024?SERVICE=WMS&VERSION=1.1")[0];
```

```
List<SubLayerInfo> subLayers = new List<SubLayerInfo>();
```

```
subLayers.Add(new SubLayerInfo() { Name = "sea_water_potential_temperature" });
```

```
subLayers.Add(new SubLayerInfo() { Name = "sea_surface_height_above_geoid" });
```

```
subLayers.Add(new SubLayerInfo() { Name = "sea_water_salinity" });
```

```
subLayers.Add(new SubLayerInfo() { Name = "sea_water_potential_temperature_at_sea_floor" });
```

```
subLayers.Add(new SubLayerInfo() { Name = "northward_sea_water_velocity" });
```

```
subLayers.Add(new SubLayerInfo() { Name = "sea_ice_velocity" });
```

```
}
```

7.5 Representation of Weather Forecast Data:

Applying the forecast layers on the top of the base map gives an overview of the weather forecast for that region. However, it won't help to predict the next day weather because applied layers give general idea about the weather information by seeing different colors in layers. So, in order help the users, numeric forecast data is provided. Through use of the previous numeric data the user can easily guess the next day weather.

7.6 Architecture of Requesting the Data from the Database

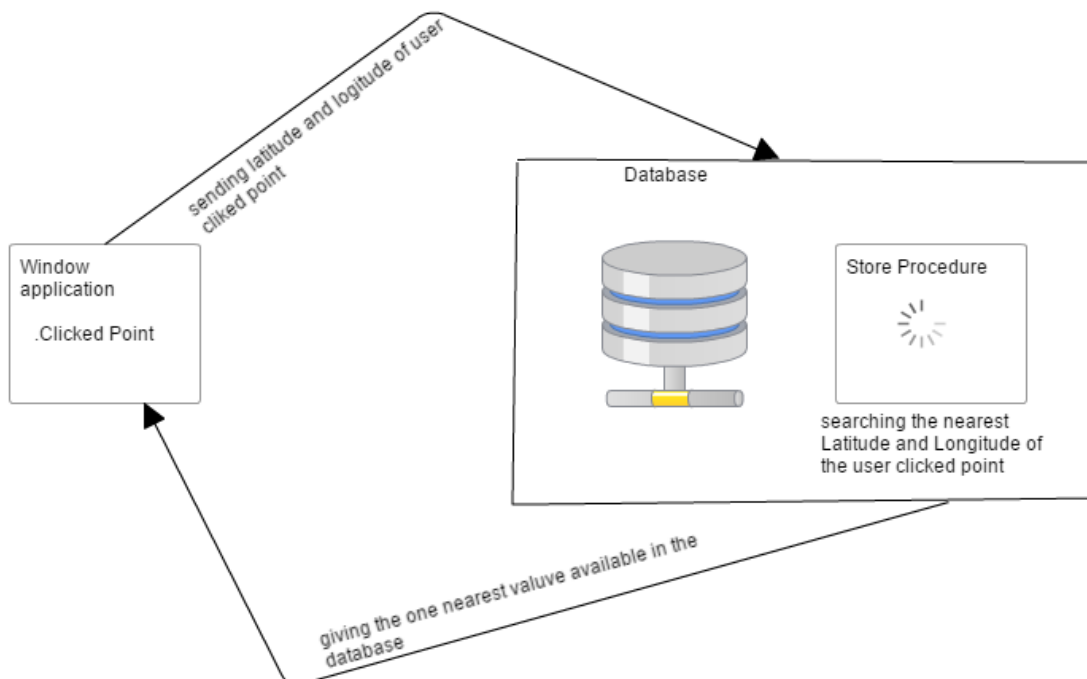


Figure 43: Architecture of requesting the data

Once the layer is applied is applied the user demands to know the numeric data value of the particular clicked point. The request contacts the database to get the values of the point using the location coordinates. The figure above shows that the clicked point of the users sending to the database. Once the store procedure receives the clicked point it will search the database of the nearest available point of the particular date and return one value for one date.

7.7 Implementation with Windows Form

The first Earth observation desktop application was implemented with windows form. The release displayed the data in the numeric form. The numeric data of the particular layers are

stored in the database, when the user clicks on the top of the layers, the window will pop up with the particular information of the numeric data of the particular user clicked point.

At the first time, the data representation is done in the Windows form which is as shown below:

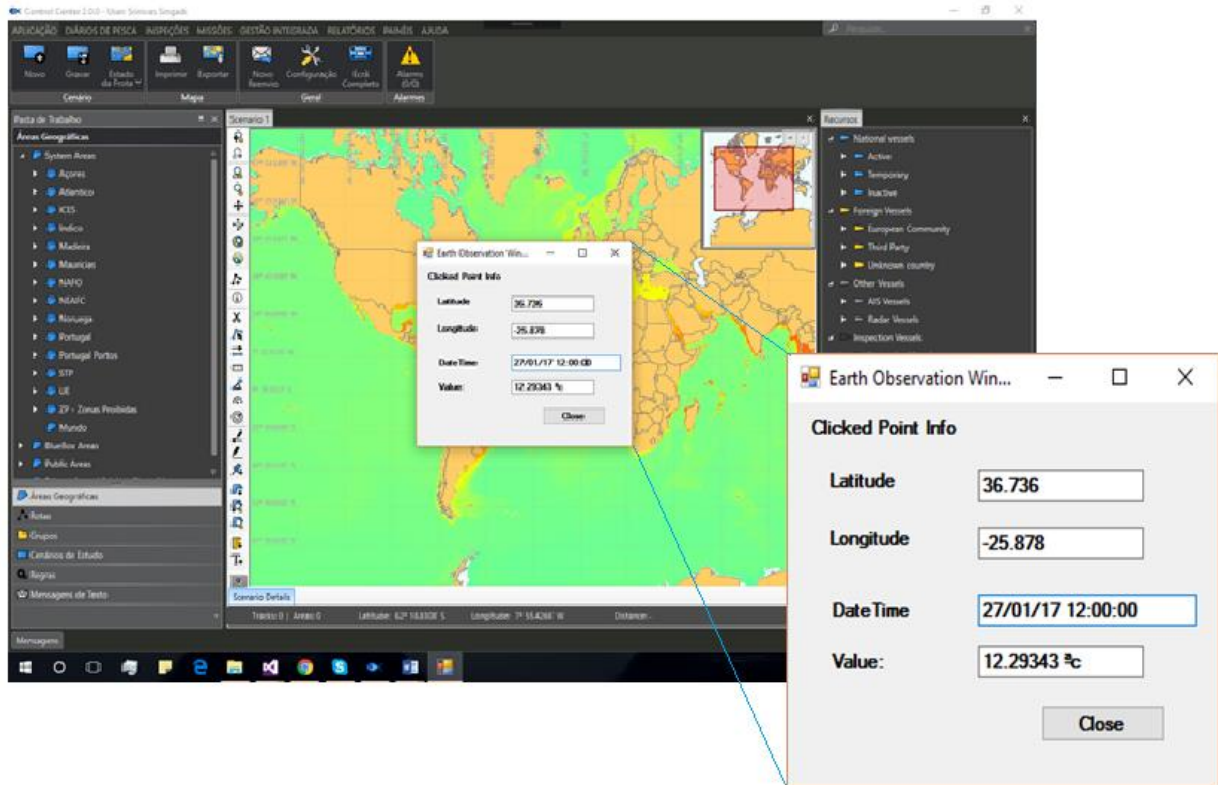


Figure 44: layers numeric data representation

The application was sent to the supervisor for review and testing. From his feedback, together with feedback from fellow users with his organization. It was noted that, from the window from implementation, users had some problems. The users could not move the map when the earth observation window pop up. Also, it was observed that the users could not see the see the previous values and with this window because in this earth observation window user can see the only one value.

As a result of analysis of this feedback, a second iteration was needed to remove these defects and ensure the application is built with a user-centered mind as the supervisor always wanted.

7.8 Implementation with Annotation

The second iteration began with a careful analysis of the feedback from the supervisor and his assistants. The user requirements which mostly focused on weather forecast representation called for a proper design of the presentation window. As a result, with the assistance of my supervisor, we adopted an annotation style of representation. In the second-type implementation, it was to help overcome the problem of the first type i.e. windows form user faced the problem that when they clicked at the particular point on the layer they can't able to move the map. Annotation representation was thought to help remove the difficulty of map movement when well implemented. The proposed changes were implemented and tested before being sent to the supervisor for further quality assurance testing. The appearance of the annotation on the map is as shown in figure 45.

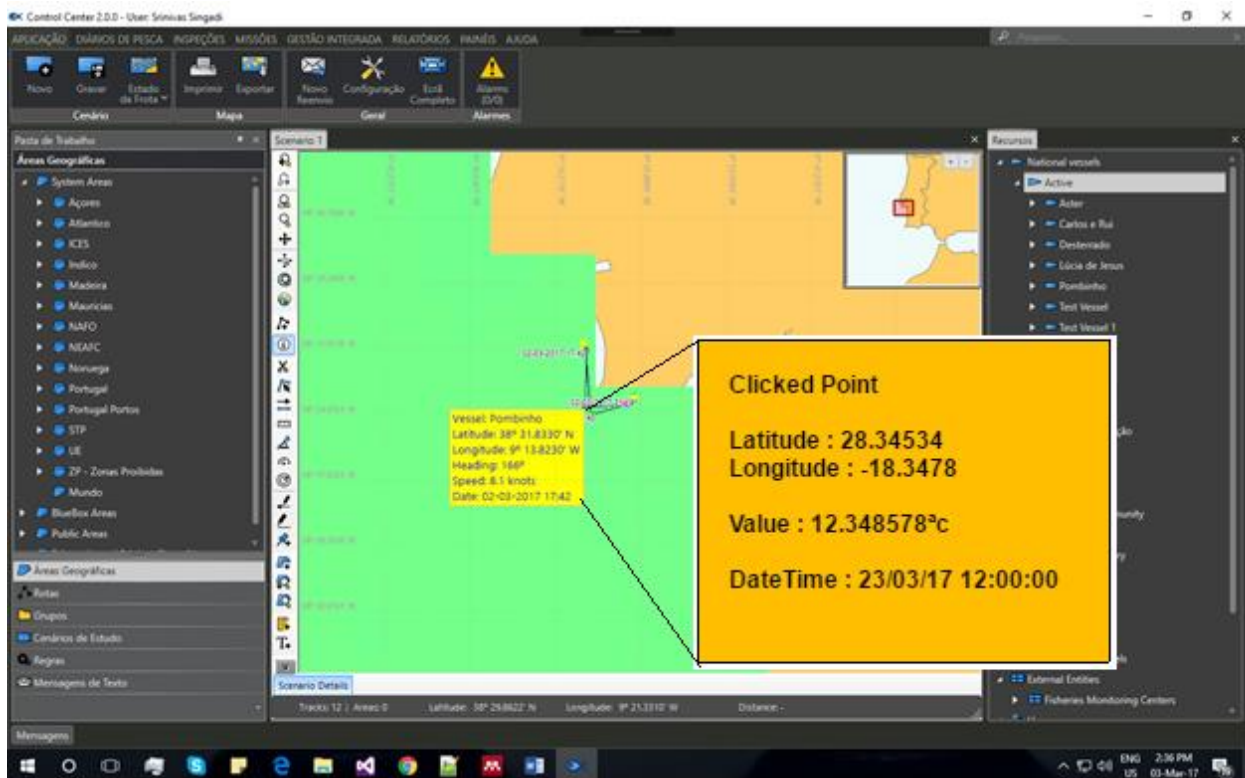


Figure 45: Annotation representation

Annotation representation of weather forecast data created a drawing on the map. However, the supervisor noted that so it is not possible to close that annotation drawing as annotation got disabled whenever the user disabled the forecast layer. Disabling the forecast layer is not good as it gives the window a bad appearance. When the users click multiple points on the maps it looks ugly, and also, it's possible to disable the previous annotation when the users click at the next point, but the last clicked annotation remains on the map. The supervisor

noted that the implemented annotation representation disturbed the other activities in the project. During a discussion on the second release, we proposed a third release to eliminate the problem. The review and testing team composed of me as the developer, supervisor and other employees proposed the use of WPF window to show the previous days value. So, the third release was to implement a WPF window with a 10-day weather data forecast.

7.9 Implementation with WPF window

WPF window helps the users to see the previous values, the implementation with WPF give the clear idea of the previous days; weather data of the clicked point. For implementing the WPF window there is slight change in the store procedure that is the store procedure returns the 10-day values. These values are not of the same date. They consist current date value and previous 9 days value of the same coordinates. These values are plotted in a graph as shown below.

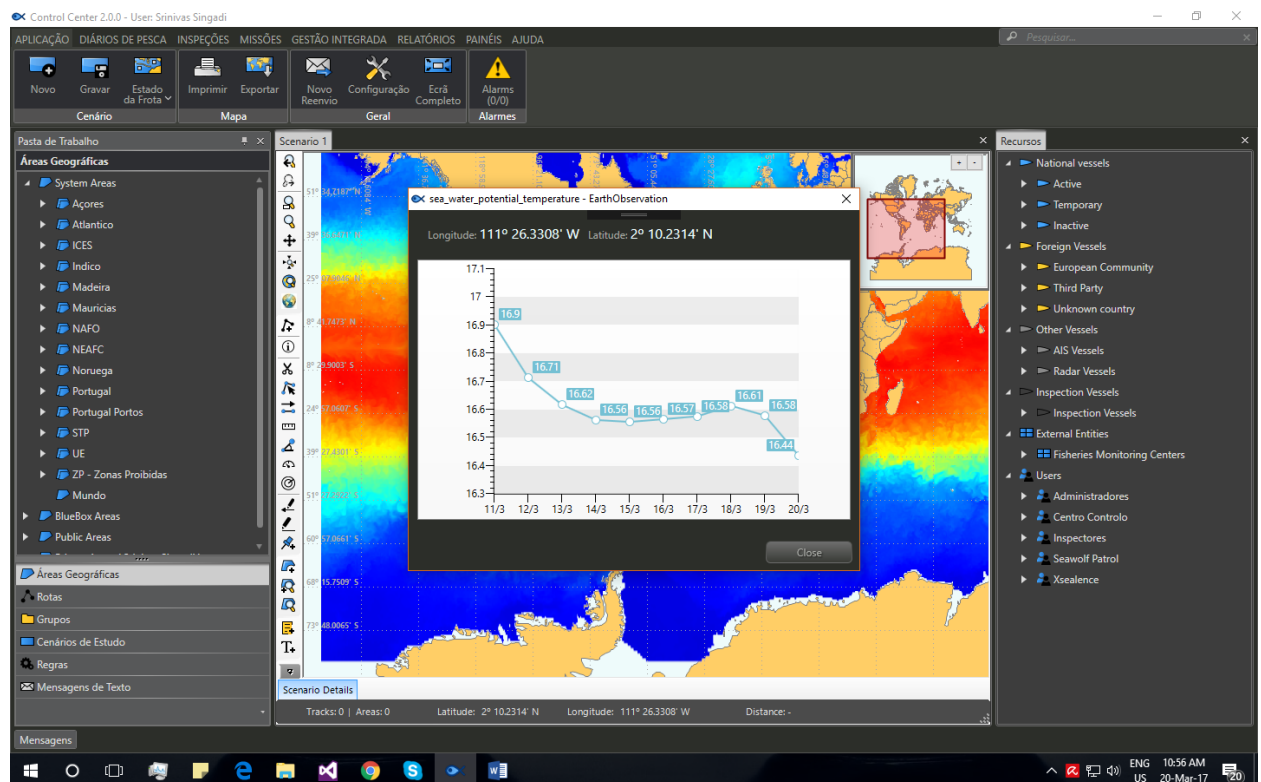


Figure 46: WPF window of numeric data

The WPF earth observation window utilizes the Telerik support to draw the graph, in the figure 46 the x-axis representing the date and y-axis representing the values.



Figure 47: WPF window with Graph representation

In the above windows, it's looking all are having different window names, because those windows belong to different layers, so the name of the window helps the users to understand which layer information they are looking, and it also window consists the coordinates i.e. latitude and longitude of the user clicked point on the map. In one of the window i.e. sea_ice_velocity-earth observation is not having any y-axis values, because the sea velocity is zero for all dates that are the reason it's showing only the x-axis values and y-axis value as zero. The main reason for providing the graph of previous days value to the users, so they can predict the next day value by observing the past days values.

7.10 System Testing

An application needs to be tested to remove errors that were not identified during development or otherwise correct usability issues. The tests that were performed served to

look for usability problems and to test the current state of application robustness. This will not be the last test phase of the Vessel Tracker since it is an application still under development. The Earth observation is the one part of the vessel monitoring project, and moreover, it was developed under the advice of company supervisor.

7.10.1 Unit Testing

Unit testing is usually done by the developer to look for syntax, runtime and compile time errors. During Earth observation system development individual units and components as well were subjected to tests so as to validate every unit [53]. The testing was also aimed at ensuring every unit works as it had been designed. Few inputs were also given to the unit to test if the output given is what was expected.

Python scripting helped identify various code errors and defects which help improve the performance of the application

7.10.2 White Box Testing

White box testing was done to test the internal working of the application coding and infrastructure [54]. Since it needed the knowledge of the internal structure of the application, white box testing was only restricted to the developer and the fellow developers within the company. The testing was primarily focused on improving the application security. The testers applied some inputs and did a process walk through to ensure the output processing was done according to standard procedure[55]. The defects found during the process walk through were corrected before the application being deemed tested.

7.10.3 Quality Assurance Testing

To ensure all project development goals and objectives are realized the supervisor helped in quality assurance testing [53]. After the supervisor analysis were discovered some errors/failures that at the time of development, and the case of lack of color consistency or something that allows the differentiation of routes. These errors will be corrected as soon as possible. With this analysis, it was also possible to collect good suggestions, ranging from small changes to complementary functionalities that will be presented to XSealence, almost all errors and issues are fixed at the time development by analyzing several times by the company supervisor and also helped to improve some part the project development.

7.10.4 Test Cases

Test Case Ref	Test Case Name	Test Case Description	Search Parameter	checks	Expected Result	Actual Result	Bug	comments
TC 001	Dash board	GUI	if the weather shows the world map with all water bodies shown	Check to see if the map shows the world map with all water bodies shown	Show the world map with all water bodies clearly show. Also, weather options table is expected	Displayed the world map as expected with zoom icons. The page also had weather options in a table on the left. Options are written against checkable boxes	none	done
TC 002	Weat her options check	Chec kbox	GUI	Check if the weather options are selectable	Once the user clicks on a checkbox against a weather option the option is selected	On clicking any option on the weather option checkbox the option was selected	none	done

TC 003	Weat her optio ns check	Chec k box	GUI	Check if you can select more than one checkbox	Once the user clicks on a second checkbox the other selected checkbox is deselected	On clicking a second checkbox the other selected checkbox is deselected	none	done
TC 004	Temp eratur e layers	Tem perat ure page	GUI	Check if on selecting the temperature weather option, the temperature weather layers are shown	Once the temperature checkbox is clicked the temperature s layers of various locations of the sea are shown using different colors to represent different colors	On choosing the temperature option from the check box screen various temperature s layers of various locations of the sea were shown using different colors to represent different degrees of temperature	none	done
TC 005	Temp eratur e coord inate	Coor dinat e pop up	GUI	Check if after display of Temperature page, if clicking on any location	On clicking a specific location on sea water, the user	On clicking a particular coordinate, that location Temperatur	none	done

		wind ow		gives the location feature information including the Longitude, latitude time and value of the clicked location	expects the see the location coordinates as well as Temperature value and time of the location	e value, longitude, latitude as well as time		
TC 006	Salinity layers	salinity page	GUI	Check if on selecting the salinity various salinity levels at various locations is shown	Once the salinity checkbox is clicked the various salinity layers of various locations of the sea are shown using different colors to represent different colors	On choosing the salinity option from the check box screen, various salinity layers of various locations of the sea were shown using different colors to represent different colors	none	done

TC 007	Salinity coordinate	Coordinate pop up window	GUI	Check if after display of Salinity page, if clicking on any location gives the location feature information including the Longitude, latitude time and value of the clicked location	On clicking a specific location on sea water, the user expects the see the location coordinates as well as Salinity value and time of the location	On clicking a particular coordinate, that location longitude, latitude, time and Salinity value were shown	none	done
TC 008	Water Velocity	Water velocity page	GUI	Check various velocity are shown with direction of water	The user expects to see the velocity of water in various locations	When the water velocity check box was selected various arrows showing the direction of water displacement were show	none	done

						all-over to map to indicate the water velocity and direction		
TC 009	Ice velocity	Ice velocity page	GUI	Check if on selecting ice velocity option shows the velocity of the ice	Once the option is selected point to the direction of ice movement	Showed the displacement of ice movement with arrows pointing from high ice content areas such as south polar.	none	done
TC 011	Surface Height	Surface Height page	GUI	Check if on selecting the surface height option various heights of the sea are shown.	The user expects to see various sea surface heights above the geoid.	Various sea surface heights above the geoid were shown upon selection of the options. These heights were represented by different layers to indicate difference height	none	done

						ranges		
TC 012	surface height coordinate	Coordinate popup window	GUI	Check if after display of Surface Height page, if clicking on any location gives the location feature information including the Longitude, latitude time and value of the clicked location	On clicking a specific location on sea water, the user expects the see the location coordinates as well as surface height value and time of the location	On clicking a particular coordinate, that location coordinate was shown as well as surface height value as well as time	none	done

7.11 Conclusion for Front-End Development System Testing

In above section discussed various test cases. And system testing check if the system conforms to the expectations of the client. Earth observation continuous testing revealed various issues that needed to be solved before going live. Presentation of the weather data was one of the issues identified. Through testing the system was evolved from windows data presentation to annotation presentation to Windows Presentation Foundation. This later presentation gave weather forecast data for the last ten days which was considered more reliable than one-day weather forecast. The data was presented in a line graph for easier understanding of weather behaviour. The functionality of each component was also tested using test cases and there identified errors corrected.

Chapter 8: Conclusion and Future Work

The project used Tatuk GIS map engine despite there being another map. This map engine was chosen since it was easy to apply the layer directly from the server. Even though Sharp map engine is easy to apply the layers directly from the server it's more expensive when compared to Tatuk map engine. Earth observation desktop application was part of the big system that was being developed which has used Tatuk for development and thus, there was no other option than to use Tatuk GIS map engine

To develop the earth observation desktop application adopted agile development methodology which was to allow parallel programming. According to my supervisor, I was to do a part of the system while the company I was attached to did the rest of the system. Agile process model allows the system to be broken down into small manageable tasks which are assigned to small agile teams for development. As a result, once the bulky application was developed in releases which help ensure the part is developed to perfection through the cyclic development process of the model. The project was developed in three releases which helped produce a final version with Windows presentation which gave the weather data forecast for the last 10 days shown in a graph.

The model guided the project development through various development phases. The phases helped collect system requirements, analyze and design the system requirements using use case diagrams, data flow diagrams as well as the use of the class diagram to show how classes will be implemented in the frontend. The project used various technologies and tools as the development platforms. Microsoft .NET version 4.5.1b was used as the development environment in XSealence. Visual Studio 2016 was used as the integrated development environment (IDE) while the Tatuk 4.6.0.9975 was used the editor to merge the base map with the weather forecast layer. Highly robust, high-performance, enterprise-class database Oracle database that allows SQL programming was used to develop project database. The database is commercial and expensive compared to another SQL database which is open source. However, its security and continuous maintenance by the Oracle community made it's the best choice for the project. To download Copernicus the backend part of the system was developed using a python scripting using Python version 2.7. The system was tested by various people who included the developer, supervisor and fellow company developers. To deploy the application Motu client python was used. The system testing included various test

cases which evaluated if the project aim and objectives were achieved and to what degree were they achieved.

As a result of developed earth observation, a desktop application which will be integrated into other application that has already developed by Company will help a lot in the management of the marine life. Earth observation application will give information about various marine weather options such as water velocity and salinity as well as temperature, ice velocity, and height of the sea surface. This means various sea and ocean users are able to use the application to help in sea navigation and fishing.

The system can be advanced and dispatched for use by various groups of people which include sea explorers, small-scale fishermen, ship navigators and other interested parties who need to access the data. However, our application does not give real time data as some apps discussed in the existing application. This means there is still room for improvement in the next iteration to try and give real-time weather data and avoid using the data that download in the previous midnight.

The weather forecast data are given if for some predetermined dates where the current WPF presentation gives marine weather prediction of the past 10 days. In the previous releases, the forecast data was only given for just one day. If a user wants marine weather forecast data for some specific dates is almost impossible to get as they have no freedom to add the dates. The future release should ensure that the user is given freedom to choose dates or days that they wish to get weather forecast data for. This will make it more useful as many users are not just interested in the present day of the last 10 days but also need weather forecast for some specific days of their own. In this case, the user should be allowed to specify “from date” and “to date” for the marine weather forecast.

Bibliography

- [1] C. Clover, *The end of the line : how overfishing is changing the world and what we eat*. University of California Press, 2008.
- [2] “Overfishing -- Pristine Seas -- National Geographic.” [Online]. Available: <http://ocean.nationalgeographic.com/ocean/explore/pristine-seas/critical-issues-overfishing/>. [Accessed: 15-Feb-2017].
- [3] “Overfishing | Threats | WWF.” [Online]. Available: <http://www.worldwildlife.org/threats/overfishing>. [Accessed: 15-Feb-2017].
- [4] R. Hilborn and U. Hilborn, *Overfishing : what everyone needs to know*. Oxford University Press, 2012.
- [5] A. de Inovação, “Portuguese vessel monitoring system sells worldwide.”
- [6] “Páginas - Homepage.” [Online]. Available: <http://www.amn.pt/Paginas/Homepage.aspx>. [Accessed: 15-Feb-2017].
- [7] “MSN Weather.” [Online]. Available: <https://www.msn.com/en-us/weather>. [Accessed: 23-Feb-2017].
- [8] H. Wu, Z. Li, H. Zhang, C. Yang, and S. Shen, “Monitoring and evaluating the quality of Web Map Service resources for optimizing map composition over the internet to support decision making,” *Comput. Geosci.*, vol. 37, no. 4, pp. 485–494, 2011.
- [9] O. G. Consortium, “OpenGIS Web Map Service version 1.3.0,” no. OGC 06-042, 2006.
- [10] Pancheng Wang, Chongjun Yang, Zhanfu Yu, and Yingchao Ren, “A load balance algorithm for WMS,” in *IEEE International IEEE International IEEE International Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004*, vol. 5, pp. 2920–2922.
- [11] K. Kommana, “Implementation of a Geoserver application for GIS data distribution and manipulation,” 2013.

- [12] A. Randall, “Web Map Service (WMS) Introduction for Science On a Sphere ®.”
- [13] “Copernicus - Marine environment monitoring service.” [Online]. Available: http://marine.copernicus.eu/services-portfolio/access-to-products/?option=com_csw&view=details&product_id=GLOBAL_ANALYSIS_FOR_ECAST_PHY_001_024. [Accessed: 03-Mar-2017].
- [14] Tom Lochhaas, “Review of PredictWind App - PredictWind App for Sailors,” *ThoughtCo*, 2013. [Online]. Available: <https://www.thoughtco.com/predictwind-app-review-2915292>. [Accessed: 12-Jun-2017].
- [15] “MarineTraffic: Global Ship Tracking Intelligence | AIS Marine Traffic.” [Online]. Available: <https://www.marinetraffic.com/en/ais/home/centerx:65/centery:43/zoom:2>. [Accessed: 03-Mar-2017].
- [16] “Home - PredictWind.” [Online]. Available: <https://www.predictwind.com/>. [Accessed: 06-Mar-2017].
- [17] “Map Results | Wind, Forecast, Radar, Weather & More | FishWeather.” [Online]. Available: <http://www.fishweather.com/map#22.792,41.832,2,7>. [Accessed: 06-Mar-2017].
- [18] Graham Averill, “The Four Best Weather Apps (Plus a Bonus for Storm Geeks) | Outside Online,” 2016. [Online]. Available: <https://www.outsideonline.com/2099756/four-best-weather-apps-plus-bonus-storm-geeks>. [Accessed: 12-Jun-2017].
- [19] Geospatial World, “A GIS application for weather analysis and forecasting,” *Geospatial World*, 2009.
- [20] “Description of GIS.” [Online]. Available: http://www.casa.arizona.edu/MPP/Um_erosion/gisdesc.html. [Accessed: 07-Mar-2017].
- [21] “What is GIS (Geographic Information System)? | Top Online Colleges.” [Online]. Available: <http://www.toponlinecolleges.com/what-is-gis-geographic-information-system/>. [Accessed: 07-Mar-2017].

- [22] “TatukGIS - Developer Kernel (GIS SDK).” [Online]. Available: <http://www.tatukgis.com/Products/DeveloperKernel.aspx>. [Accessed: 14-Mar-2017].
- [23] “TatukGIS - Description Editor and free Viewer (GIS Desktop).” [Online]. Available: <https://www.tatukgis.com/getdoc/a52b07d7-a0ee-4410-a103-fb5ac0085916/Description.aspx>. [Accessed: 04-Jun-2017].
- [24] “TatukGIS - Editor and free Viewer (GIS Desktop).” [Online]. Available: <http://www.tatukgis.com/Products/EditorViewer.aspx>. [Accessed: 14-Mar-2017].
- [25] “SharpMap - Geospatial Application Framework for the CLR - Documentation.” [Online]. Available: <https://sharpmap.codeplex.com/wikipage?title=Tutorial - Example 3%3A Adding a WMS-layer to the map&referringTitle=SharpMap Tutorial>. [Accessed: 14-Mar-2017].
- [26] M. A. Cusumano and S. Smith, “Beyond the Waterfall: Software Development at Microsoft Section 2: SOFTWARE DEVELOPMENT PROCESSES AND PROBLEMS.”
- [27] Y. Bassil, “A Simulation Model for the Waterfall Software Development Life Cycle,” *Int. J. Eng. Technol.*, vol. 2, no. 5, pp. 2049–3444, 2012.
- [28] “Waterfall Model Definition | IT & Systems Dictionary | MBA Skool-Study.Learn.Share.” [Online]. Available: <http://www.mbaskool.com/business-concepts/it-and-systems/8658-waterfall-model.html>. [Accessed: 29-Mar-2017].
- [29] A. Alshamrani and A. Bahattab, “A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model.”
- [30] Er Parag Verma, “SDLC Incremental Model design phase, applications, advantages and disadvantages | Engineer’s Portal,” 2015. [Online]. Available: <http://er.yuvayana.org/sdlc-incremental-model-design-phase-applications-advantages-and-disadvantages/>. [Accessed: 16-Jun-2017].
- [31] V. Wariyal, S. Gupta, and V. Kumar, “Process Models in Software Engineering,” *Int. J. Innov. Res. Eng. Sci. ISSN*, no. 10, pp. 2319–5665, 2013.
- [32] Dele Oluwole, “Scrumhint | Agile Software Development,” *Scrunhint*, 2016. [Online].

- Available: <http://www.scrumhint.com/snapshot-of-agile-software-development/>.
[Accessed: 16-Jun-2017].
- [33] Neelam Gupta, “Lectures 2 & 3: Software Process Models Neelam Gupta. - ppt download,” .
- [34] “Copernicus - Marine environment monitoring service.” [Online]. Available: <http://marine.copernicus.eu/about-us/about-eu-copernicus/>. [Accessed: 26-Jun-2017].
- [35] “About CAMS | Copernicus Atmosphere Monitoring Service.” [Online]. Available: <https://atmosphere.copernicus.eu/about-cams>. [Accessed: 17-Jun-2017].
- [36] “Copernicus - Marine environment monitoring service.” [Online]. Available: <http://marine.copernicus.eu/services-portfolio/access-to-products/>. [Accessed: 14-Mar-2017].
- [37] “Visual Studio | Developer Tools and Services | Microsoft IDE.” [Online]. Available: <https://www.visualstudio.com/>. [Accessed: 10-Apr-2017].
- [38] “Welcome to Python.org.” [Online]. Available: <https://www.python.org/>. [Accessed: 10-Apr-2017].
- [39] “Motu Client Python download | SourceForge.net.” [Online]. Available: https://sourceforge.net/projects/cls-motu/?source=typ_redirect. [Accessed: 10-Apr-2017].
- [40] Systems Analysis and Design - Harry J. Rosenblatt - Google Books. .
- [41] Structured System Analysis and Design - J. B. Dixit - Google Books. .
- [42] Modern Systems Analysis and Design - Joseph Valacich, Joey George - Google Books. .
- [43] I. Govender, “From Procedural to Object-Oriented Programming (OOP) - An exploratory study of teachers’ performance,” *Res. Artic. — SACJ*, no. 46, 2010.
- [44] Joydip Kanjilal, “Exploring Association, Aggregation, and Composition in OOP | InfoWorld,” 2016. [Online]. Available: <http://www.infoworld.com/article/3029325/application-development/exploring->

- association-aggregation-and-composition-in-oop.html. [Accessed: 15-Jun-2017].
- [45] Understanding UML Class Relationships and Vaughn Vernon, “Understanding UML Class Relationships,” 2004. [Online]. Available: https://vaughnvernon.co/?page_id=31. [Accessed: 15-Jun-2017].
- [46] G. Price, “Review: Oracle Essentials: Oracle 9i, Oracle 8i and Oracle 8,” *Comput. Bull.*, vol. 43, no. 6, pp. 31–31, Nov. 2001.
- [47] “What is Oracle Database (Oracle DB)? - Definition from Techopedia.” [Online]. Available: <https://www.techopedia.com/definition/8711/oracle-database>. [Accessed: 15-Mar-2017].
- [48] Help Net Security, “SQL injection has surfaced as the no. 1 attack in 2015 - Help Net Security,” 2015. [Online]. Available: <https://www.helpnetsecurity.com/2015/12/11/sql-injection-has-surfaced-as-the-no-1-attack-in-2015/>. [Accessed: 16-Jun-2017].
- [49] Ericka Chickowski, “SQL Injections Top Attack Statistics,” *Dark Reading*, 2010. [Online]. Available: <http://www.darkreading.com/risk/sql-injections-top-attack-statistics/d/d-id/1132988?> [Accessed: 16-Jun-2017].
- [50] Edgescan, “2015 Vulnerability Statistics Report,” 2015.
- [51] Oracle, “Cost Effective Security and Compliance with Oracle Database 11g Release,” 2011.
- [52] “Fast nearest-location finder (store finder) for Oracle.” [Online]. Available: <http://www.plumislandmedia.net/oracle/fast-nearest-location-finder-oracle/>. [Accessed: 22-Mar-2017].
- [53] Aaron Marcus, *Design, User Experience, and Usability: Interactive Experience Design: 4th ...* - Google Books. 2015.
- [54] N. Y. Khaled El-Fakih, Gerassimos Barlas, “Testing Software and Systems: 27th IFIP WG 6.1 International Conference ... - Google Books,” 2015.
- [55] J. (John) Mantas *et al.*, *Quality of life through quality of information : proceedings of MIE2012*. IOS Press, 2012.

