VDR WEBSITE APPLICATION TO VISUALIZE PRODUCTION DATA FOR OIL AND GAS INDUSTRIES

GIS APPLICATIONS IN THE VDR

THESIS

PROJECT

by

Chan Elizabeth W 2201797001

BINUS INTERNATIONAL

BINUS UNIVERSITY

JAKARTA

2022

VDR WEBSITE APPLICATION TO VISUALIZE PRODUCTION DATA FOR OIL AND GAS INDUSTRIES

GIS APPLICATIONS IN THE VDR

THESIS

Proposed as a requirement for obtaining
Sarjana degree at
Program Computer Science
Education Level Strata-1 (Sarjana/Bachelor)

by

Chan Elizabeth W 2201797001

BINUS INTERNATIONAL
BINUS UNIVERSITY
JAKARTA
2022

Table of Contents

List of Figures

List of Tables

CHAPTER 1
INTRODUCTION

In this chapter, the background of this thesis is described, and the author and team's scope of doing the thesis is described. The objective, such as the aim and benefits, are also included in the chapter.

1.1 Background

Oil and gas were used for lamps and lubricants starting back in 346 AD [1]. The oil well drilled was done by China [1]. Then moving forward to the late 18th century and early 19th century, it was an era in which oil and gas industries that still dominate the world until today were established [1]. Since the mid-1950s, the oil demand has become the world's most important source of energy. Refined oil products are used for many things, such as supplying energy to the power industry, heating homes, providing fuel for vehicles and airplanes, and coating pills [2]. Aside from that, oil and gas are also economically essential as the oil and gas industries reduce unemployment as the industries provide numerous jobs. Oil and gas industries in America also support the country economically as the industries are projected to provide $1.6 trillion in federal and state tax revenue between 2012 and 2025 [3]. The tax revenue is then intended to support schools, hospitals, and public infrastructure across the country [3].

As for Indonesia, the well-known beneficiaries of oil and gas are vehicle fuels, Liquefied petroleum gas (LPG), and petrochemical products such as synthetic rubber and polyurethane materials [4]. Due to these beneficiaries, the oil and gas consumption increased from 1180 thousand barrels a day to 1750 thousand barrels a day in the span of 8 years from 2000 to 2018 [5]. Unfortunately, these beneficiaries also lead to decreased oil and gas production, causing lesser income for the country [5]. However, the oil and gas industries had provided an enormous contribution to state revenue of around 216.9 trillion rupiahs in 2014, which boosted the country economically [5]. According to *Badan Pusat Statistik* (BPS) Statistics Indonesia, there were 20,326 Indonesians that worked in the oil and gas industries in 2019 [6]. The unemployment number is lesser with these numbers, representing more economic growth.

With all the importance and benefits of oil and gas and the decrease in oil and gas production mentioned earlier, many software applications such as Virtual Data Room (VDR) now help the oil and gas industries. The VDR is an online data room that requires authorization to access it. VDR helps the industries to store and analyze data. It also attracts potential clients to invest in the industries by giving clients the platform to visualize the data. Other software applications may also help the industries to analyze which locations can produce more oil and gas. Software with a visualization tool can also help oil recovery [7]. For data visualizations, several well-known software services are Schlumberger's ProSource Front Office, Lynx, and INTViewer. On one side, Lynx provides visualization through seismic viewers. While, INTViewer, provides visualization of seismic and geospatial data. These software services are used by oil and gas industries in Indonesia. For example, Pertamina utilizes INT's web-based HTML5Viewer with their platform to visualize their data quickly and efficiently [10]. With the high demand, the author and team were encouraged to develop a VDR application similar to the well-known software mentioned earlier. Thus, the primary goal of this thesis is to develop the VDR application with customizable features as requested by the author's client, PT Geodwipa Teknika Nusantara (GTN) [11]. The goal was backed up by the product owner, Mr Ardimas Andi Purwita, as he stated that some oil and gas companies in Indonesia hope that there will be a customizable software similar to Lynx and INTViewer [12]. In addition, the proposed VDR application will be developed as a web application as part of the requirements from the author's client.

The customizable features requested by the author's client include implementing a Geographic Information System (GIS) in the VDR. A GIS is a system of software, hardware, and data that enables the exploitation, analysis, and display of data and information based on the location on the surface of the Earth that is connected to the data [13]. It helps to enhance the VDR application by providing a better understanding of mining locations through visualizations, determining which locations are better in terms of use, and better geographic data documentation for the industries and clients [13]. The knowledge of the data is richer after being analysed using GIS [13]. Data is also easier to search, explore and display with a GIS map [13].

Other than GIS feature, the other customizable features that the author's client requested are visualization of oil and gas data, prediction model to predict oil and gas production, and file management which is stated in Figure A.2. However the author is assigned to the GIS feature so this thesis will concentrated on:

- developing map visualization of oil and gas production data and
- developing map with showcase feature.

1.2 Scope

The scope of work that the author and the author team are responsible for and individual's scope are explained below.

1.2.1 Team Scope

In this thesis, the author and team are responsible for assembling different kinds of visualizations, and their diverse functionalities merged in the end into a website application. Each team member had separate tasks that required them to successfully finish their tasks one by one to create a complete website. The complete website will have various features including:

- uploading and storing files and showcase information,
- deleting and downloading files,
- downloading showcase information,
- data visualizing and data showcasing with maps, and
- acquiring oil and gas production data with the use of a predictive model.

To complete the tasks, the author's team decided to implement an agile methodology as the Software Development Life Cycle (SDLC) in developing the tasks. The agile methodology is chosen due to certain benefits discussed in Section 2.5. The tasks and roles in the SDLC for each member can be seen in Table 1.1.

Table 1.1 Scope of Tasks

Member names	Tasks
Chan Elizabeth W	• Design and develop all features of the website application related to GIS including the oil and gas visualization that integrated GIS and a custom showcase of oil and gas wells data on the map showcase page. • Design and develop the form pages where user need to input the data for visualization and showcase. • In the agile methodology, her role is as a scrum member.
Kotrakona Harinatha Sreeya Reddy	• Evaluating different machine learning algorithms on oil and gas dataset to find the ideal model. • Deploying the best model as an API to be used by the VDR Website application for oil and gas prediction. • In the agile methodology, her role is as a scrum member.
Vicky Vanessa	• Designing UI of the frontend of the website application. • Visualizing the data of oil and gas. • In the agile methodology, her role is as a scrum master.

The out-scope work of the author and team include storing and organizing data in the database. These works are done by the backend developers from Pertamina University.

1.2.2 Individual Scope of Work

In this thesis, the author is responsible for the front-end of the features utilized with a GIS. The author has responsibility of designing the looks of the map pages that implement GIS inside the website, developing the map pages based on the design, and implementing user interaction. In detail, the author will develop the map to be shown, the components to mark the location of the data available, and features such as sort and filter that provide the users with efficiency in visualizing the data. In addition, the author is also responsible for developing a custom showcase. The showcase will display general information regarding oil and gas based on the chosen locations for analysis that can be easily understood by non-technical users or other users that are in a hurry to read basic information. By being responsible, the author is determined to use the frameworks that can provide new experiences of GIS that provide efficiency for the VDR users.

1.3 Aims and Benefits

The aims and benefits of developing the VDR website application are described in this section.

1.3.1 Aims

The team's aim and individual's aim are discussed in this subsection

1.3.1.1 Team Aim

The general aim of developing this VDR website application is to help the oil and gas industries by:

- providing customizable features inside the VDR website application,
- visualizing the reserve resources, and
- developing predictive model that will help the oil and gas industries predict and reveal potential sites containing more oil and gas.

1.3.1.2 Individual Aim

As not many existing VDRs have implemented GIS, the author has personal aim to:

- integrate GIS features inside the VDR website application,
- present data showcase in the GIS feature
- visualizing the reserve resources based on the location of the oil and gas wells

1.3.2 Benefits

The major benefits of VDR website application and the benefits of the author's aim in this subsection.

1.3.2.1 VDR Benefits

The VDR website application benefits oil and gas industries to:

- personally store and visualize their data, and
- attract many domestic and international clients.

The application is also useful to aid engineers in comprehending complex data and understanding how the data can be used. With the application, the author and team also hope to increase the use of local services in the oil and gas industries that help the industries to gain more profits and indirectly increase the national income.

1.3.2.2 Individual Benefits

Based on the individual aim mentioned in Subsection 1.3.1.2, the benefits of developing the feature that the author responsible of are:

- to provide new experiences for VDR clients to efficiently visualize the data with the integration of GIS and
- to allow VDR clients to be more easily determine which assets are most profitable according to data showcased in the GIS feature.

1.4 Structure

This thesis consists of five chapters that are shortly explained below.

1.4.1 Chapter 1: Introduction

Chapter 1 introduces the background, the scope, and the objective, including the aims and benefits of this thesis.

1.4.2 Chapter 2: Theoretical Foundations

Chapter 2 describes and explains the theoretical concept of the approaches in which the thesis is develop with and the tools used by the author to develop the thesis.

1.4.3 Chapter 3: Problem Analysis

Chapter 3 discusses the difference between existing application and proposed solution. The reason why the author chooses to use the particular tools and frameworks to make the proposed solution are also explained.

1.4.4 Chapter 4: Solution Design

Chapter 4 further explains the user interaction with the system, the flow of the system, and the interaction of the thesis features along with the design.

1.4.5 Chapter 5: Implementation

Chapter 5 explains how the system is implemented in developing the VDR website application.

1.4.6 Chapter 6: Discussion

Chapter 6 provide discussion regarding the process of developing the VDR website application.

1.4.7 Chapter 7: Conclusion and Recommendations

Chapter 7 will conclude all the results of developing the VDR website application and explain the recommendations for future work.

CHAPTER 2
THEORETICAL FOUNDATION

This chapter explains the theories behind the proposed application. The theories include the things that the author wants to develop, such as VDR, GIS, the web application, and front-end development—then followed with the theories on open-source software as the author used this kind of software in developing the application. After that, theories on the SDLC of developing the application are described.

2.1 Data Room (Virtual data room)

As one of the functionalities of this thesis is to store data and visualize it through a website application, the thesis implements the concept of a Virtual Data Room (VDR). The VDR is a kind of data room. By definition, a data room is a precious instrument when oil and gas companies want to sell their assets [15]. With data room, companies can attract more buyers by providing information for buyers to consider their products [15]. Aside from VDR, the other kinds of data rooms are Physical data rooms (PDR) and a combination of PDR and VDR [15].

The PDR is a room with high surveillance [15]. The room is filled with confidential documents that the team of potential buyers will examine thoroughly to find the financial secrets of potential investments [15]. As there are many documents, it is a time-consuming, expensive, and complex activity [15]. It is time-consuming due to analyzing and sorting data that are useful [51]. It is expensive due to the need to spend on high surveillance rooms and traveling if buyers' and sellers' origins are different [15]. It is complex due to the need to copy extensive important data and ship it back to its origin [15]. It is also complex because the sellers need to keep the data up-to-date until the buyers have time to visit the sellers' PDR [15].

Due to the complexity and significant expense of PDR, it is largely replaced by VDR [15]. The VDR is an online data room that requires authorization to access it [15]. With authorization such as unique passwords for buyers to access the VDR, it is as secure as PDR [15]. Aside from security, VDR also advances accessibility and saves costs [15]. For sellers, it is easily accessible for quick data updates [15]. At the same time, potential buyers can easily access the website from any location and at any time [15]. Buyers also gain the advantage of saving money by not needing to travel [15]. Therefore, it can be concluded that VDR is better than PDR.

Some of the examples of VDR are FirmRoom and Lynx's GeoViewer. FirmRoom provides a files and documents management page on their VDR website where users can upload, store, organize, and view the files and documents, as shown in Figure 2.1 [16]. While Lynx's GeoViewer provides visualization of the data stored on the website, which is chosen based on the data locations on the map, as shown in Figure 2.2 [17].

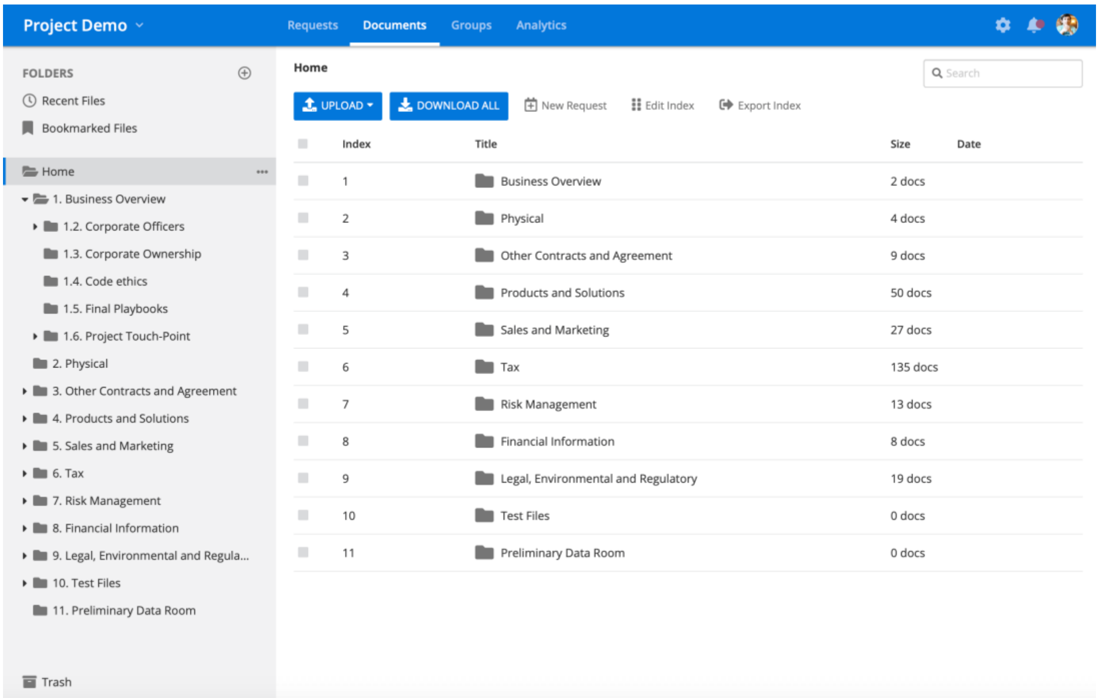


Figure 2.1 FirmRoom's Data Room Index Demo [16]

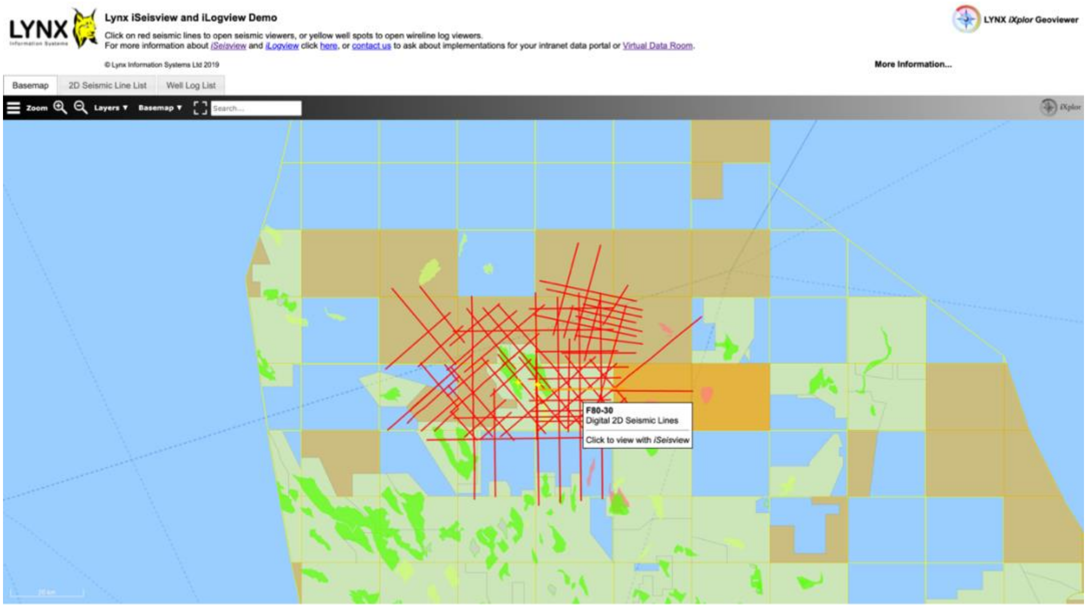


Figure 2.2 Lynx's Geoviewer Demo with seismic viewers and well log viewers [18]

2.1.1 vtkPolyData (VTP)

The data store inside the VDR can be in different file formats [19]. Data can be in form of standard files such as PDFs, spreadsheets, and word processing documents [19].

Data can also be in form of videos, pictures, or anything else based on industries that use VDR [19]. For example, VDR for oil and gas industries like Lynx's Geoviewer may requires storing of 3D geophysics objects like seismic data, well logs, and surface data. This 3D geophysics objects are mostly represented using Visualization Toolkit or VTK Polydata (VTP) format. In definition, VTP is a file format to store VTK model [20]. VTK alone is a software for rendering and making 3D computer graphics and visualization [21].

2.2 GIS

One of the examples of VDR mentioned in Section 2.1 followed the concept of GIS to visualize the data chosen based on the locations of the data on the map. GIS is a system of software, hardware, and data that enables the exploitation, analysis, and display of data and information based on the location on the surface of the Earth that is connected to the data, as mentioned in Section 1.1 [13]. There are many other definitions of GIS established by different viewpoints. However, GIS can be understood by analyzing the three letters of the acronym GIS, which is explained below [13]:

- The letter 'G' represents Geographic which indicates an attraction to a terrestrial location of a particular substance on, beneath, or beyond the surface of the Earth.
- The letter 'I' represents Information that indicates the importance of data to be managed into helpful information that helps in decision-making.
- The letter 'S' represents a System that indicates the necessity for a team, computer hardware, and procedures (data gathering, processing, and visualization) to produce the information that helps in decision-making.

Up to this point in time, GIS has undergone many adaptations, which are [13]:

- Paper Mapping spatial analysis with Cholera Clusters where Dr. John Snow, a British physician, started mapping the locations of Cholera outbreaks, including the roads, surrounding buildings, and waterlines, in 1854. The mapping was done by putting geographic layers on a paper map. When he mapped these features, he distinguished that Cholera cases were commonly found along the waterlines. The map represents the significant relation between geography and public health safety with these findings.
- Before 1960, it was known as the Dark Ages of GIS due to computer mapping was not developed, so all mapping was done on paper. All the features are drawn on paper which can be complicated for cartographers.
- 1960 to 1980s is known as the pioneer time of GIS due to the transformation of mapping from paper to computer.
    - In the 1960s, Roger Tomlinson, known as the father of GIS, collaborated with the Canadian government, where he originated, arranged, and led the Canadian Geographic System (CGIS) development. The CGIS was defined as the roots of Geographic Information Systems by many people. CGIS was unique because it adopted a layer approach system to map handling. The CGIS is used to measure which Canadian lands are suitable for planting and cropping based on data of soil, drainage, and climate characteristics. Over the years, CGIS has been modified and improved to keep pace with technology.
    - In 1970, the United States Census Bureau started to analyze/count data digitally using a format that implements the main principle of GIS known as GBF-DIME (Geographic Base File – Dual Independent Map

Encoding). Other than the ability to input data to be stored, the format also supported error fixing and choropleth mapping. Using this format, the US Census Bureau began to digitize the number of borders, streets, and metropolitan areas. Data stored digitally was a huge step forward in the history of GIS.

    - During this period, the Ordnance Survey in the UK also started to develop a topographic map regularly, and it continues till now.

- From 1975 to 1990 is known as GIS software commercialization, where GIS software vendors increased. Esri, the largest GIS software company globally, started to provide GIS services. The other things that happened during this period were conferences and published work which also increased the acknowledgment of GIS.
- From 1990 to 2010, companies or individuals started implementing GIS for different purposes. Although in the beginning, companies still hesitate to implement GIS. The GIS began to be recognized by people as a valuable tool that helps in analyzing and decision-making. The software was also able to handle both vector and raster data throughout the time. The data collected from space could also be implemented in GIS as more satellites are being launched into orbit.
- From 2010 till now, people have been introduced to open-source GIS. With GIS that leads open-source GIS, people who have computers are given the freedom to use GIS software. Therefore, GIS is becoming more popular, and more advanced open-source GIS is being made.

With all the transformations of GIS throughout time, the GIS always provides advantages such as [13]:

- better decision making on what the locations are useful for,
- improved communication as GIS maps and visualizations provides a better understanding of the locations,
- better geographic data documentation,
- the knowledge of the data is richer after being analyzed using GIS,
- questions of the analyst regarding the geographic data can be answered quickly by the GIS visualization,
- in a GIS environment, layers can represent different data,
- specialized maps or visualize tools can provide prediction results based on data,
- GIS eliminates the multiple redundant maps set,
- map data is easier to search, explore and display, and
- cost savings resulting from greater efficiency.

Nowadays, GIS is mainly in the form of the web. The standard features for web-based GIS are shown in Figure 2.3 [40]. In the Figure 2.3, triangles represent special GIS features, crosses represent non-GIS features, and diamonds represent special web mapping features [40].

Figure 2.3 Probable standard features of a web-based GIS grouped by functionality [21]

2.2.1 Online Map Services

In the present day, most web users use maps and geospatial data on a day-to-day basis. Web maps are issued by several national mapping agencies [41]. Online map services are broadly used for navigation, locating desired places, travel arrangements, concurrent traffic details, education, and many others. [41]. Online map services publicize maps with raster or vector data along with satellite images and provide features for interactivity [41]. The features include the ability to zoom in and out of the map, the ability to pan the map, retrieve information, the ability to navigate, and many others. [41]. Some online map services, such as Google Maps, can be used worldwide. In contrast, some online map services, such as Baidu Maps, can only be used in China [41]. As there is more demand for online map services from web users, the providers of map services are competing with each other to improve their services [41]. The criteria of online map services to be distinguished as worldwide popular are if they stand in high ranking, cover globally, and have many users worldwide [41]. According to November 2018 map website with the most traffic in the world provided by Similarweb (www.similarweb.com), the online map services that fulfill the criteria above include Google Maps, OpenStreetMap, Here maps, and Wikimapia [41]. The traffic data of the online map services are shown in Table 2.5.1.1 [41].

Table 2.1 The traffic data [41]

Online Map Services	Monthly Visits in Millions (August 2018-October 2018	Percentage of Visits from Country with Greatest Popularity	Year Founded	Main Traffic Source on Desktop	Rank in the Similarweb on map website with the most traffic	Global Rank	Platform Used by Most People
Google Maps	108	USA (30.04%)	1998	Direct	1	691	Mobile
OpenStreetMap	10	Germany (14.72%)	2004	Direct	6	1819	Mobile/Desktop
HERE Maps	16	USA (19.23%)	1985	Social	8	2021	Desktop
Wikimapia	9	Russia (36.76%)	2006	Search	10	2692	Mobile/Desktop

For better information, a definition for each online map service above is defined below [41]:

- Google Maps: It is developed by Google. The features of Google Maps include maps, 360◦ streets views, satellite imagery, actual-time traffic situations, and a route guide for traveling in various ways (foot, trains, and other ways). Most Android phones have Google Maps as the default map app.

- OpenStreetMap (OSM): It is an open-source platform developed to provide the world with a free customized map. It originates from affordable portable satellite navigation instruments created against expensive and restricted access to worldwide geographic information platforms. OSM is categorized as Volunteered Geographic Information (VGI). Furthermore, OSM also allows the creation of custom maps from the vector geographic data provided.

- HERE Maps: It is developed by HERE Technologies. The company was known to have government clients and famous clients such as BMW, Amazon.com, and Facebook. The products sell or are licensed to the clients, including content for mapping, navigation services, and location services for GIS.

- Wikimapia: It was created by Alexandre Koriakine and Evgeniy Saveliev. It works as a geographic encyclopedia project where all worldwide geographical entities are marked and described in an interactive map.

2.2.2 GIS Libraries

As online map services mentioned in Section 2.2.1 increase in demand, the development of web GIS technologies is also trending [40]. More and more open-source or commercial web-based GIS frameworks and libraries are available [40]. The most widespread web-based GIS frameworks are ESRI’s ArcGIS and CARTO’s CartoDB.js [40]. Following the nature of the web, web-based GIS software also consist of server-side and client-side components [40]. There are three types of architecture for the components: thin client architecture, thick client architecture, and medium client architecture [40]. Most web-based GIS frameworks used thick client architecture, which had a concept that the client-side is responsible for rendering the raw spatial data [40]. However, it also had a rich server-side component with geoprocessing functionality [40]. Other than frameworks, the famous JavaScript web-based GIS libraries are Leaflet, Cesium, NASA Web World Wind, and Open Layers [40].

The Leaflet is a lightweight library with a shallow learning curve (the measurement of understanding level when developers use the library) [40]. It has extensive third-party extensions [40]. The examples of the extension are Vue2Leaflet and vue2-leaflet-geosearch. Vue2Leaflet is an extension that acted as a wrapper library for the Leaflet by providing Vue components that wrap almost all the Leaflet's functionalities [42]. While vue2-leaflet-geosearch is a Leaflet.GeoSearch (library for geocoding) plugin extension for Vue2Leaflet library [43]. In addition, Leaflet is suitable for projects requiring a competent Web GIS where extensibility is not required [40]. The Leaflet is also beginner-friendly as it has detailed documentation and a good support team [40].

Cesium is a library with lesser coverage of GIS features [40]. It is categorized as a virtual globe [40]. It is suitable for a project expected to deliver a complex result while it starts to be a simple project [40]. This is due to the potential increase of the learning curve throughout the time of development [40]. The developer may need to explore and learn a lot by themselves as Cesium only provide essential documentation [40].

NASA Web World Wind is a lightweight library alternative of Cesium that allows the transformation of vector data accurately between global and eight polar projections [40]. The 2 of 8 projections supported by NASA are also supported by Cesium [40]. It is also categorized as a virtual globe [40].

Open Layers libraries include Open Layers 2 and Open Layers 3. These libraries are suitable to support heritage browsers [40]. It is also suitable for applications with statistical abilities as the libraries can have the functionality of searching and filtering [40]. The GIS features of Open Layers 3 are more thorough rather than other libraries, and the documentation is quite complete, so the learning curve is intermediate [40].

2.3 Website application

The VDR including the GIS features will be built as a web application. Therefore, it is vital to understand more about website applications and their development process. The development of websites has also rapidly evolved since the first-time worldwide web was proposed by Tim Berners-Lee in 1989 [22]. The first purpose of the World wide web was to allow worldwide scientists to share information virtually. Aside from that, three essential technologies formed by Tim as the foundation of the web until nowadays are Hypertext Markup Language (HTML), Uniform Resource Identifier (URI), and Hypertext Transfer Protocol (HTTP) [23]. HTML is the formatting language for documents displayed in the web browser [23]. URI, also known as URL, is a unique naming system that identifies each resource on the web [23]. HTTP is the application layer protocol responsible for interacting users with web resources and transferring information between devices connected to the internet [23]. After Tim formed the foundation of the web, he made and deployed the first web page, which was simple and static without any pictures and primary fonts [22]. It was believed to be the only type of web page at that point in time. Fortunately, as time passes, many different design types of websites (collection of web pages) play an essential role in website development.

One of the common types is the static website which is similar to the first web page that Tim made, but it is less demanded as website technology and purposes have advanced [22]. The other common types are dynamic, responsive, content management system (CMS)-built, and eCommerce [22] [24]. A dynamic website is the upgrade of the static website where the web application's contents are editable, the website has many functionalities, and the user can interact with the website [22]. The only limitations of the dynamic are setting up the functionalities can be complicated, and the website loads slower due to the compositions and elements present on each webpage [24]. This type of website is the active website that is available nowadays. The other type of website that is followed by many websites these days is responsive. It is famous as the responsive website can be accessed seamlessly from different devices, including mobile phones, and the website can adjust itself to fit the browser size [25]. A responsive website provides a fantastic experience for the user and increases productivity with this ability. However, there are still limitations such as browser compatibility issues when opening the website from the old browser version, slow load, and complications in developing the website considering perfect user experience [25]. Moving on to CMS-built websites, it is websites that allow non-programmers to add, modify and delete website pages along with their contents, such as texts and pictures [22]. A famous example of CMS-built websites is WordPress. The last common type of website is the eCommerce website that sells services and goods that customers can buy directly as the website is connected with a payment gateway [22].

In order to develop any types of websites mentioned above, developers need to understand the web application architectures. The web application architecture is the base of all components of the websites [26]. Developing the web application will be time-effective and cost-effective if the base is concrete [26]. The components of web application architecture are divided into two components which are structural (server) components and UI and UX (client) components [26]. The server component includes the web application's server and database [26]. At the same time, the client component provides for the layout of the elements of the websites, input controls, dashboards design, and many others [26]. These components are the general compositions of the system [26].

Aside from components of web application architecture, the layers of web app architecture are responsible for dividing the websites into layers to allow easy upgrade or modification of each layer independently [26]. The three layers of web app architectures are the presentation layer, business logic layer, and persistence layer [27]. The presentation layer is exposed to users using a browser and contains the user interface display and process components that allow communication with the system [27]. Next, the business logic layer, also known as the application layer, has the role of getting the user requests from the browser, processing the requests, and choosing the paths from where the data will be retrieved [27]. This layer encodes the functions of commutating the data and requests to the backend [27]. The last layer, the persistence layer, also known as the storage or data access layer, collects all the requested data and gives access to the application's data storage [27]. The persistence layer is linked to the business layer, so the logic recognizes which data storage has the requested information and enhances the process of retrieving it [27].

Some famous web application architecture patterns include the client-server architecture, Microservices, serverless, progressive web apps, and single-page applications [26].

- The client-server architecture pattern is the most common architecture where the client sends a request to a server, and as a response, the server will send data packets to the clients [26].
- The microservices architecture pattern is based on the concept of using an independent microserver to be responsible for a function in the system. As there can be many micro servers, each is connected with one other. This pattern is suitable for complex web applications as adding, maintaining, and testing the functionality of different parts of the system is straightforward [26].
- The progressive web apps (PWA) architecture pattern provides an advanced user interface for clients. These advanced features include running the web application offline, delivering push notifications for clients, and connecting with hardware's APIs [26].
- Single-page applications (SPA) architecture pattern allows clients to access all information from a single HTML page. With this pattern, the server loads are reduced significantly as only a component of the page area is reloaded based on clients' requests [26].

As there are many different architecture patterns, developers shall choose wisely. The website is guaranteed high performance, security, and trustworthiness with the right and excellent architecture pattern.

2.3.1 Front-end development

Front-end development is the part of web development that concentrates on developing the presentation layer of web app architecture. It also involves converting the backend developers' code into a graphical interface and ensuring that the data displayed is reader-friendly and understandable [28]. In addition, front-end developers need to ensure that the websites work in various device types, OSs, and different web browsers. Front-end developers need to consider that the website shall work on client applications, such as Chrome, Safari, Microsoft Edge, Internet Explorer, Mozilla Firefox, and Opera.

When people work as front-end developers, there are three technologies that they need to master. The three technologies are HTML, Cascading Style Sheets (CSS), and JavaScript [38]. As discussed before, HTML is the main component of a website that acts as a formatting language for documents to be displayed in the web browser. CSS is a set of instructions accountable for the website's style, such as colors, animations, and layouts [28]. HTML and CSS are enough to create websites, but JavaScript is used to make the websites interactive. However, in the past, website performance is low. Therefore, Google presented the Chrome V8 JavaScript engine in 2008 that performs a lot faster than Internet Explorer [29]. JavaScript has excellent performance after the V8 engine was presented, similar to Java [29]. Therefore, the website can run rapidly. Because of the V8 JavaScript engine's high capabilities during that time, many JavaScript frameworks were formed afterward, which evolved the internet development [29]. Some JavaScript front-end frameworks available today are React Js and Vue Js.

React Js work by distributing webpage content as various components in the Document Object Model (DOM), and then the browser generates the component using JavaScript [29]. The advantage of React Js over traditional HTML websites is that React Js only re-render the part of the website that is updated rather than re-render all the parts of the website, which is time-consuming [29]. React Js can do so by comparing data bind with virtual DOM and displayed DOM. While Vue Js also uses DOM binding to update the content appearance of the website. The only difference between React Js and Vue Js is that React Js depend on functional programming principles where libraries manage state and communication between components [29]. In contrast, Vue Js provides built-in features and supporting libraries to make the development experience seamless [29]. However, there are particular abilities that the Vue Js are more formidable than React Js, such as optimization efforts and familiar beginner-friendly templates [30]. Better optimization efforts as there is a function in the Vue component that is automatically implemented to know which components need to be re-render when state changes, while React Js require the developers to implement the function first or it will not be implemented [30]. Vue Js provide HTML-based templates that allow developers with experience in HTML to quickly adapt, while React uses JavaScript and JSX (an XML-like syntax) that function in JavaScript [30]. Vue also allows developers to implement CSS in *<style>* component familiar with HTML component [30]. With Vue, the HTML, CSS, and JavaScript components can be written in a single file and known as Vue Component [30]. With these advantages, Vue Js is an excellent choice for developers, especially newbies.

2.3.2 Vue Js

Vue.js is a famous JavaScript open-source front-end framework invented by Evan You in 2014 [29]. The initial purpose of Vue being developed is to provide reactive data binding and UI components using an uncomplicated Application Programming Interface (API) [29]. Although a single-page application has limited functions and is hard to implement in commercial use, Vue can operate complex single-page applications, including state management, routing, and build tooling, with the support of third-party libraries and packages [29].

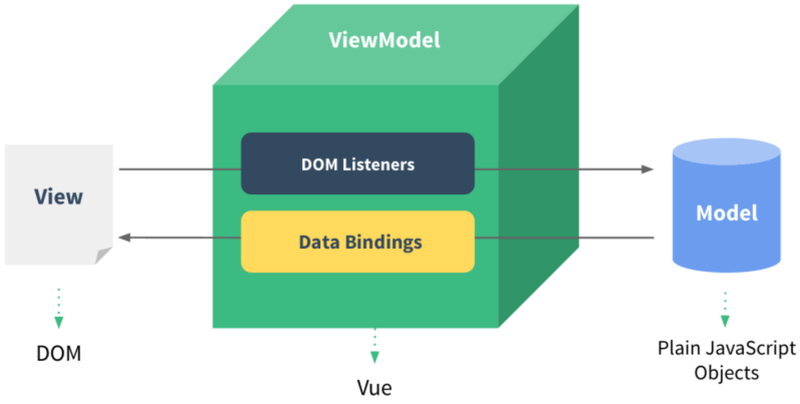


Figure 2.4 Vue Js data-driven concept [31]

Figure 2.4 above demonstrates that Vue.js retains three units to process data-driven, including View, View Model, and Model. The view unit is the displayed DOM that contains the website content [29]. View Model unit contains Data Bindings and DOM listeners. It becomes a middleware responsible for the transmission between View and Model. Then, Vue uses the DOM listeners in the View Model unit to observe and update the data in the Model unit when users begin the data processing in the View unit [29]. After the data in the Model unit is updated, the DOM binding is used to update the display of the website content [29]. Therefore, it can be concluded that the one-way binding with DOM listeners in Vue has the same ability as two-way binding [29].

2.3.3 Front-End Design Frameworks

In order to make the front-end of the web application attract visitors, excellent design needs to be implemented. Therefore, this thesis's Front-End design frameworks are Materio, Material Design, Vuetify, and Material Design Icons. Materio Admin is an open-source, user-friendly, and customizable admin template built with Vue Js and Vuetify [32]. It was created by ThemeSelection and came out in two versions [32]. Both versions of Materio are utilized with Vuex, Vue Router, Webpack, and Material design icons [32]. The features of the free version include [32]:

- a vertical menu,
- a dark and light skin layout,
- a dashboard,
- basic pages,
- basic cards,
- basic tables,
- a chart library,
- basic documentation,
- regular support, and
- manual customization.

While the features of the professional version include [32]:

- a collapsible vertical menu and horizontal menu,
- three dark and light skin layouts: Default, Bordered, and Semi-dark,
- three choices of dashboards,
- five API ready applications,

- advance form elements, validation, and the form wizard,
- basic cards, advanced cards, and statistics cards,
- advanced tables,
- advanced charts, including two chart libraries,
- two choices of authentication pages and various operable pages like login page, pricing, FAQ, knowledge base, and others,
- multiple navbar and menu options,
- 3D characters and Illustrations,
- detailed documentation,
- starter-kit,
- access control, including the function CRUD,
- quick search enables fast navigation between pages using hotkey support,
- internationalization support in components, and
- priority support.

With only the features of the free version, developers can create attractive and exclusive single-page applications. Therefore, the author and team chose the free version as the project template. The other advantage is that the free version uses the MIT license, the most straightforward license, which is declared free-forever [32]. The free version is also available to be run in the latest browsers such as Chrome, Safari, Firefox, and Opera [32].

The Vuetify is used to build the Materio, and the Material design icons that are utilized in Materio are part of the material design. Google developed material design to create a united style for all webs and mobile apps [33]. In definition, material design is a flexible design of guidelines, components, and tools to support user interface design

[33]. Many different material design frameworks and libraries offer components and styles that are directly usable [33]. One of the material design frameworks is Vuetify. At the same time, one of the material design libraries is Material design icons.

Vuetify is a UI framework. The purpose of Vuetify is to provide developers with tools that help them create engaging user experiences [34]. Unlike other frameworks, Vuetify is designed from the very beginning to be developer-friendly [34]. It is developed based on Material Design specifications and thoroughly prepared for every component [34]. The features provided by Vuetify include Accessibility, Bidirectionality (LTR/RTL), Breakpoints, Global configuration, Icons, Internationalization, Layouts, Presets, Programmatic scrolling, SASS variables, Themes, Tree shaking, and Weekly code release on fixing issues found by the community [34]. Additionally, Vuetify provides 18 months of long-term support for each major release [34]. When these features are compared with other Vue UI frameworks, as shown in Figure 2.5, Vuetify has the most thorough features, so it is recommended to use Vuetify.

Vue Framework Comparison 2022

Features	Vuetify	BootstrapVue	Buefy	Element UI	Quasar
Accessibility and section 508 support	●	●	●		
Business and enterprise support	●				
Long-term Support	●				
Release cadence**	Weekly	Bi-Weekly	Bi-Monthly	Bi-Weekly	Bi-Weekly
RTL support	●	●		●	●
Premium themes	●	●			
Treeshaking	Automatic	Manual	Manual	Manual	Automatic

***Based on average of all Major/Minor/Patch releases over the last 12 months.*

Figure 2.5 Vue Framework Comparison 2022 [34]

Material Design Icons is a library containing icon collections that can be downloaded in any format, color, and size according to the designers' or developers' projects' needs [35]. It can be utilized in various frameworks, including Vue and React [35]. It is maintained by Austin Andrews, and the general license of Material Design Icons is Pictogrammers Free License [35]. However, some icons are redistributed with Apache 2.0 license while others are redistributed with their respective license [35]. All fonts are also redistributed with Apache 2.0 license, and all non-font and non-icon files in the Material Design Icons GitHub apply the MIT license [35]. Material Design Icons can be downloaded using npm or from the website [35].

2.4 Application Programming Interface

Application Programming Interface or API is an intermediary that allow third party applications to access data or services of a software application without the need of implementing the essential objects or procedures [36]. The third-party applications can be various type of applications including website applications. There are four steps on how an API works [37]:

1. The third-party application trigger an API request to get the data,

2. The API process the request to the web server if the request is valid,

3. The web server deliver the response to the API along with the desired data, and

4. API forward the response to the third-party application that made the request.

The request by the third-party application is normally make using the API's Uniform Resource Identifier (URI) along with the request headers, verbs, or a request body [37]. As the use of APIs increased, there are different type of API protocols that accommodate regulated information exchange [37]. One of the common protocols used is REST API which is the API that follow the design principle of Representational State Transfer (REST) architectural style [38]. REST APIs provide communication to be done via HTTP requests to operate database functions such as create, read, update, and delete records (CRUD) within a resource [38]. All HTTP methods shall be useable in API calls [38]. In the API calls, the request headers and parameters are also important because they contain important identifier information [38]. With a valid request, the REST APIs will return the response information in any format including JavaScript Object Notation (JSON), HTML, Python, PHP, or plain text [38]. Aside from the response information, the response headers and response HTTP status codes are also return to the third-party application [38].

2.5 JavaScript Object Notation

JavaScript Object Notation also known as JSON is a lightweight text format for storing and transporting data [39]. JSON is popular format used for response information from the web server to the REST APIs and to the application that requested [38]. It is popular as it is readable by both humans and machines [38]. It is also ideal as it is an independent language with a familiar convention of the C-family of languages (C++, Java, JavaScript, Python, and many others) [39]. There are two universal data structures that JSON is built on which are a set of name/value pairs and an ordered list of values [39]. A set of name/value pairs are commonly known as an *object*, dictionary, or associative array in different programming languages [39]. While an ordered list of values are commonly known as list, array, sequence, or vector [39].

2.6 Open-Source Software

With all the frameworks and libraries mentioned in Section 2.2.2 and Section 2.3.3, open-source software (OSS) is popular to use. Open-Source Software (OSS) is software that gives liberty for anyone to review, alter, and enrich the source code [44]. OSS has ruled the world as nowadays virtual product, service, or platform is powered by it [45]. Aside from using OSS, many companies have acknowledged the benefits of OSS and started to create platforms for the source code development of the OSS. Companies such as Google established Google Code in 2005, which supports OSS developers with fundamental tools and allows them to share their projects with the public [44]. Afterward, Microsoft released CodePlex, which allowed engineers and computer scientists to share their ideas and OSS projects, although Microsoft was known to oppose Open Source [44]. In 2018, Microsoft also acquired GitHub, the most prominent open-source development platform to this time [45]. Since then, Microsoft has become the world's most significant open source contributor based on the number of employees actively contributing to open source projects on GitHub [46].

Whether companies provide their software as OSS, companies contribute to OSS, or developers use OSS, there are many benefits. The benefit for companies making their software as OSS is reducing their development cost and innovation cost of the software as their software can receive contributions from talented developers around the world [47]. While the benefits for companies that contribute to OSS are attracting excellent developers to join their companies without spending money on recruiters and attracting developers to use the paid upgraded version of OSS as they might feel unquestionable loyalty after using the free version for some time [48]. In addition, the benefits for developers in using OSS are the development can be cost-effective, and they are given freedom and flexibility to explore the OSS and produce new unique ideas [49]. OSS is also secure for use as developers participating in the OSS constantly review code, build improvements, and fix issues [49]. Aside from that, developers that participate in the OSS can increase their problem-solving skills by being exposed to different coding styles and new ways of fixing issues [48]. Developers also can increase their communication skills by communicating with other developers with different backgrounds that participate in the same OSS [48].

OSS is preferred over proprietary or closed source software with all the benefits explained above. The term proprietary software is used to define software that only the stakeholders of the software have the right to copy, inspect, and modify the software along with the source code [44]. To use proprietary software, users must sign a license declared the first time they open the software to agree that they will avoid doing anything with the software that its authors have forbidden [44]. As users need to sign a license to use proprietary software, users also need to accept the terms of a license when they use the OSS. The terms often describe how people can use, study, modify and distribute the OSS [50]. As a requirement, the terms inside open-source licenses shall retain the criteria established by Open-Source Initiative [51]. The criteria, which are also known as Open-Source Definition, include [52]:

- free redistribution,
- source code of the software shall also be distributed and provide free access,
- the license shall allow modifications and work based on existing works. The license also needs to state the approval of the modified software to be distributed under the same conditions as the license of the original software,
- the integrity of the author's source code,
- the license shall not discriminate against any person or group,
- the license shall not discriminate the purpose the software is used,
- distribution of license,
- the license shall not be specific to a product,
- the license shall not restrain other software, and
- the license shall be technology-neutral.

Based on the criteria above, it can be concluded that open-source licenses do not limit the user's purpose of using the OSS in general. However, there are open-source licenses that require anyone who redistributed a modified program of an OSS to attach the source code and preserve the same rights as the original; these licenses are called "copyleft" [44]. An example of the same rights that need to be preserved by the redistributors is they are not allowed to charge licensing fees for the redistributed modified OSS. The well-known copyleft license is GNU's General Public License (GPL). When developers use GPL features in their software, they need to obey the terms and conditions below [53]:

- on the software, developers are not authorized to claim patents or copyright. Furthermore, developers must include and show intact GPL notices, a copyright notice, a GPL copy, and a warranty disclaimer on the software,
- it is forbidden to alter the license or add extra terms and conditions, and
- developers are bound by the reciprocal agreement that obligates them to release the software's source code and provide others with the right to modify and redistribute.

Aside from that, software with a GPL license can be sold by the users whether the software is still original or already modified [53]. However, GPL gives buyers of the software the right to release the software to the public with price or not [53].

Other than GPL, which is categorized as copyleft, the famous open-source licenses that are not categorized as copyleft licenses are BSD licenses, MIT licenses, and Apache licenses. All three licenses are categorized as permissive software licenses, which acquire minimal restrictions on how the software can be used, altered, or redistributed either for open source or proprietary by the users [54]. Although they are categorized as one, they have their own terms requirements. When users use BSD licenses, they need to keep the copyright notice, the disclaimer, and the limitation on all the redistributed source codes and documentation supporting the redistribution [55]. When promoting the software with BSD licenses, users also need to include written permission from the contributors to write their names in the copyright notice [55]. While the MIT license is identical to BSD, MIT allows the user to include the name of the contributors in the copyright notice for promotion purposes [55]. On the other hand, Apache is quite different as the license agreement is long to provide clear concepts on how the software can be used and a description of patent rights [56]. Almost all the open-source software that the author and team used has an MIT license and Apache license.

2.7 Software Development Life Cycle

In order to successfully develop the website application more structural and faster, a Software development life cycle (SDLC) is used. SDLC is a set of processes utilized to develop, design, and maintain software projects and ensure that the result of the projects corresponds with the user requirement [57]. The processes of the SDLC, also known as phases, which are necessary for the project's development to be completed include requirement analysis, design, implementation, testing, and deployment and maintenance/evolution [58].

- Requirement analysis [58]

  It is the first phase of SDLC where the development team needs to grasp the detail of the client's requirement, document the requirement, and determine ways to apply the requirement.

- Design [58]

  In this phase, the development team usually requires developers to be creative and utter a solution on how to develop the software and system based on the requirement specification. The risks, project limitations, development period, budget, technologies to be used, and project's subtasks distribution are discussed in this phase.

- Implementation [58]

  In this phase, the development team implements all the client's requirements. Project member such as front-end developers starts developing an interactive Graphical User Interface (GUI) that satisfy the client's requirements and connect the GUI with the APIs. At the same time, backend developers start coding the application logic, web services, and software APIs that front-end developers will use.

- Testing [58]

  It is the last phase before the software is deployed and delivered to the client. Testing is carried out to determine whether the software works expectedly based on the design decision in the earlier phase or not. With thorough testing, high-quality software can be delivered and avoid many bugs in production, which means fewer maintenance costs.

- Deployment and Maintenance/Evolution [58]

  This is the final phase of SDLC, where the software will be deployed according to client use. Then a maintenance team is usually available to monitor and report any post-production issue to the development team. The issue, if available, might require a hotfix (develop a solution and deploy in a short time) or can be fixed in the next version of the software based on the severity of the issue.

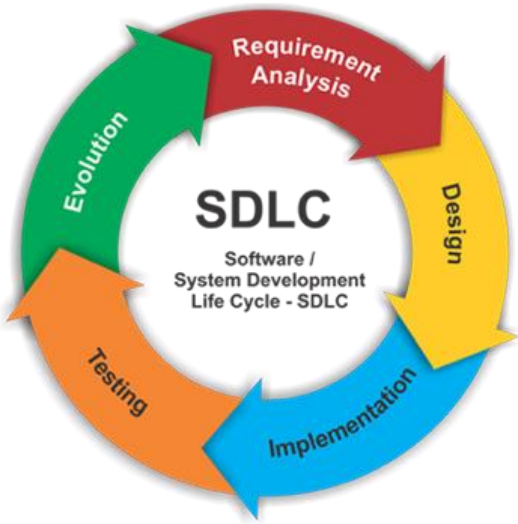


Figure 2.6 SDLC Phases [59]

Figure 2.6 shows the SDLC phases. The phases of SDLC can be undertaken slightly differently based on the SDLC models chosen to develop the software projects. Each model has different advantages and limitations when dealing with and considering certain conditions such as budget, project timeframe, and project requirements [60]. In addition, the SDLC models can be classified either as traditional models or agile models. A famous example of the traditional model is the waterfall model [61]. While some of the most popular agile models are extreme programming (XP) and Scrum.

In order to have a better understanding of the general difference between the traditional models and agile models, Table 2.2 is used to visualize the difference easily.

Table 2.2 Difference between traditional models and agile models [61]

Traditional Models	Agile Models
Planning as a control mechanism	People’s feedback as a control mechanism
In the early phase of the project, the requirements are exact.	In the early phase of the project, the requirements are still approximate.
Concentrating on documentation, less communication.	Communication is vital to develop the project in a short time. Documentation is negligible.
Follow detailed process	Limited process involved
Develop a complete project in a single cycle which takes a long duration	The iterative model allows development quickly in a short time
No scrum calls and stands up calls	Many scrum calls. Scrum calls are stand-up calls that happen recurrently to give updates on progress and receive feedback from clients
Understanding the project, developing, testing, and issue fixing takes a long time	Frequent contact with people involved in the project, iterative development, and accurate feedback have saved more time in each phase
Automation is not a usual approach	Continuous automated testing is recommended, and the use of it can assure better quality

From Table 2.2, it is clarified that agile models are better than traditional models. When teams or companies implement agile. they need to follow certain principles, which are [62]:

- customer is God, so the customers’ requirements need to be prioritized through early and continuous development of the software,
- sudden changes are regular and need to be obeyed if it benefits the customer even though development is about to finish,
- update and deliver operating software frequently in various timespan but preferably short time span,

- developers and involved business individuals shall update each other's progress daily throughout the project,
- choose the trusted individuals that can be passionate about the project and companies to provide the support and environment that the individuals need,
- face-to-face communication is the best and most effective way of delivering information between individuals in a development team,
- published and useable software is the primary goal of the project,
- agile processes facilitate endurable development where all the stakeholders of the project can preserve a steady pace in doing their jobs,
- agility is improved if developers pay attention to technical superiority and designers pay attention to great design,
- simplicity is the best policy, which is essential to reduce unfinished work,
- a self-organizing team can gather outstanding requirements, produce good designs and architectures, and
- during the daily meeting, each team member reflects on how to improve effectiveness, and then the member tries to adjust based on the things reflected.

By applying the principles, teams or companies can believe that the developed products will publish. Therefore, the author and team also decided to follow an agile model. The agile model chosen is Scrum.

Scrum is an agile model that implements iterative processes that result in significant project progress from each process [61]. This model is generally used to overcome ambiguous conditions and tight deadlines. Most of the time, there are three roles when following the Scrum model: product owner, development team, and Scrum master [63].

The events to be done when the scrum model is implemented are product backlog, daily Scrum, sprint, sprint planning meeting, sprint backlog, and sprint review meeting [63].

- Product backlog

  This is the first event where investors and project members gather all the business requirements and technical requests and list all the activities that will be developed during the project.

- Daily Scrum

  It is a quick daily meeting where project members give updates on the progress of the previous day's tasks, and it is also a time when the Scrum master assigns new tasks.

- Sprint

  It is a period of one to four weeks where the development team start coding all the tasks listed when the Product Backlog.

- Sprint planning meeting

  It is the meeting where the project members plan the things that need to be done in a sprint.

- Sprint backlog

  It is a subgroup of Product Backlog where activities that need to be done during a sprint are listed down.

- Sprint review meeting

  It is an after-sprint meeting where project members reflect on what was achieved, what does not go with the plan, and the things that can be improved.

From the events, the advantages of Scrum are defined as productive communication among project members through the daily Scrum and constant feedback from product owners, which allows delivering the products accurately as expected [64]. The only disadvantage is Scrum is relative hard to master for people with no experience with Scrum [64].

In order to perform the events with ease, Jira Software is used as it has a comprehensive set of agile tools. The Jira Software is known as number 1 software development tool used by agile teams, it is made and maintain by Atlassian (an Australian software company) [65]. The tools include backlog page and scrum board page present in the Jira application [65]. In the backlog page, scrum master and teams can estimate stories, accustom sprint scope, and re-prioritize issues in real-time [65]. In the scrum board page, all the tasks in the current sprint can be easily visualize and track in different columns based on the task's status [65]. Other than backlog and scrum board page, Jira also provide reports such as a Cumulative Flow Diagram (CFD). The CFD is useful for team to evaluate if the project progress of the team is stable, too many members in the team or the team is assign with more tasks than they can handle. Aside from Jira, Discord can be used for better communication during Daily Scrum, Sprint planning meeting, and Sprint review meeting events. Discord is well known as a voice, video and text communication service that ease many people in online hang out and talk with each other.

2.8 Design

Design is the second phase in SDLC which requires the development team to be creative and utter a design of the software and system based on the requirement specification as mentioned in the previous section. The design can be in the form of diagrams to visualize and illustrate the system. The frequently used diagrams for design the system are Unified Modeling Language (UML) and flowcharts. Aside from that, there are also a set of diagrams to illustrate the design of application's user interface. This set of diagrams are known as Wireframe.

2.8.1 Use Case Diagram

A UML diagram is a modelling method by visualization as a tool to illustrate a system and how the users can use the system [66]. There are thirteen UML design techniques: an object diagram, class diagram, package diagram, use case diagram, component diagram, state diagram, a composite structure diagram, timing diagram, activity diagram, and interaction overview diagram [66]. Each diagram represents a different perspective but corresponds with one another [66].

Use case diagram, one of the UML design techniques, is often used to help developers to visualize the functionality of the system from the users' point of view [67]. The scenarios on how users interact with the system are illustrated [67]. The notations in the use case diagram are actor, use case, association, and system [67]. The actor represents the system's user, and it is illustrated with stickmen [67]. The use case represents a function of the system, and it is illustrated with ovals [67]. The association represents the relation between the actor and the use case, and it is illustrated with lines [67]. The system is illustrated with a rectangle [67].

2.8.2 Flowchart

On the other hand, a flowchart is a diagram that illustrates a computer system, algorithm, and process [68]. With a flow chart, processes from various fields can be easily recorded, examined, arranged, recovered, and understood [68]. It is the most frequent diagram used by technical or non-technical fields [68]. Flowcharts use symbols such as ovals, trapeziums, rectangles, and diamonds to represent the type of steps and arrows to represent the direction of flow [68]. The ovals symbol defined the start and end of the flow. The trapeziums symbol defines the input or output. The rectangles symbol defined a process. The diamonds symbol defined decisions (yes or no). The other common symbol is rectangle with vertical lines on the right and left sides [69]. This symbol represents the predefined processor which mark another set of process that are defined elsewhere [69].

2.8.3 Wireframe

As mentioned earlier, wireframe is a visual guide that represents the skeletal framework of a website is often used in designing the UI [67]. With wireframe, the developers, designer, and client can visualize the entities, pages, and components the application is going to have and how these elements of the digital product will interact with each other [70]. Therefore, client can also give feedbacks to the designer whether the application will be developed already fulfill the requirement or not in early phase [71].

2.8.4 Sitemap

A sitemap is an outline of a website that assist search engines discover and crawl to all the website's content [72]. Sitemap also inform search engines on which pages of the website are the most important [72]. There are four main types of sitemaps which are normal Extensible Markup Language (XML) sitemap, video sitemap, image sitemap and news sitemap [72].

- The normal XML sitemap is commonly in form of XML sitemap that links to different pages of the website [72].
- The video sitemap mostly contain the video running time, age-suitable rating, and video rating [73].
- The image sitemap can include the location of the images on the page of the website [73].
- The news sitemap can contain the publication date and article title [73].

CHAPTER 3
PROBLEM ANALYSIS

This chapter will declare the problem that persuaded the author and team in making the solution to the problem. In this chapter, the differs between existing applications and the proposed solution will also be stated. The existing applications and proposed solution, as mentioned in Chapter 1, will be described further.

3.1 Problem

As mentioned in Chapter 1, the major problem conferred by product owner and the author's client that led to the creation of the proposed solution is the desired for a customizable application. The customizable application shall also have certain features that fulfil the requirements of the client. The requirements are stated in Figure A. 2. Specifically, the author's problem is to be responsible of developing the first and second features stated in the requirements. The features that the author need to provide solution to are:

- Visualization of oil and gas data with the integration of GIS and
- map application along with the showcase features

These features are important to be develop because GIS provide benefits for the oil and gas industries as mentioned in Section 1.3.2. Without GIS, some problems can arose such as:

- users will find it hard to understand the oil and gas well locations,
- determining which locations are better in terms of use will be harder and
- data can be harder to search, explore and display without a GIS map.

3.2 Existing Applications

This section further describes the existing applications that integrated GIS which become the sample to the GIS integrated in the VDR website application. The existing applications, as mentioned in chapter 1, include Lynx's Virtual Data Room, INT's IVAAP, and Schlumberger's Prosource Front Office.

3.2.1 Lynx's Virtual Data Room

Lynx Information Systems has been operating since 1989 [74]. It offers geological and GIS services, software solutions, and data management for oil and gas industries worldwide [74]. They provide services that include VDR. Their VDR is designed to display oil and gas analysis datasets that are useful for farm-outs, data commercialism, and acquisition chances [17]. The Lynx's VDR is secure as only authorized users can access the data from anywhere using any web browser [17]. To be an authorized user, the user needs to sign confidential agreements issued by the issuing authority with the help of Lynx staff [17]. The user's activities are recorded and accessible to the issuing authority [17].

When authorized users login to the VDR, they will see a map or list-based interface that displays certain data such as 2D seismic lines, 3D seismic, Wireline logs, and Well reports/seismic processing reports [17].

- 2D seismic lines displayed using Lynx *iSeis*view. The functionality includes the ability to select the display style, scale, and amplitude, with overlaid horizon/fault interpretations and intersecting lines/wells/3D surveys [17].
- 3D seismic is also displayed Lynx *iSeis*view using in the form of inline and crossline profiles with an interactive selection of display style, scale, and amplitude, with intersecting 2D lines/wells/other 3D surveys [17].
- Wireline logs displayed with Lynx *iLog*view in the form of interactive composite log display, select vertical scale, turn individual curves on and off, show formation tops and depth-correlated core images [17].

- Well reports, seismic processing reports in PDF documents [17].

These data need to pass quality control to be uploaded to the VDR. The user shall not be worried that the data is terrible as experienced Lynx staff will do the quality control and upload [17]. The Lynx VDR website is as shown in Figure 3.1.

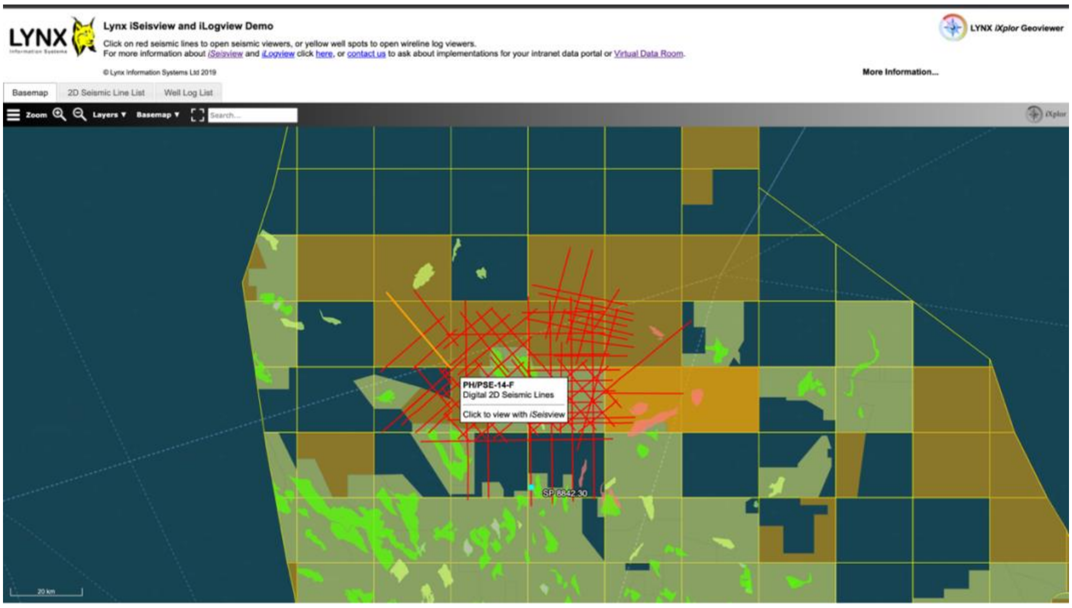


Figure 3.1 Lynx VDR Demo [18]

3.2.2 Interactive Network Technologies (INT) ’s IVAAP

INT, similar to Lynx, has helped oil and gas industries visualize, monitor, and analyze their data since 1989. The product by INT includes INTViewer mentioned in previous sections. Other products are IVAAP and GeoToolkit [75]. One of the features of IVAAP has been integrated with GIS which allows users to visualize their subsurface data on a map along with the ability to filter data to be shown on the map [76]. The IVAAP also provides the ability to connect machine learning to the selected subsurface data [76]. IVAAP also allows easy connection to the source of the data with an API and SDK [76].

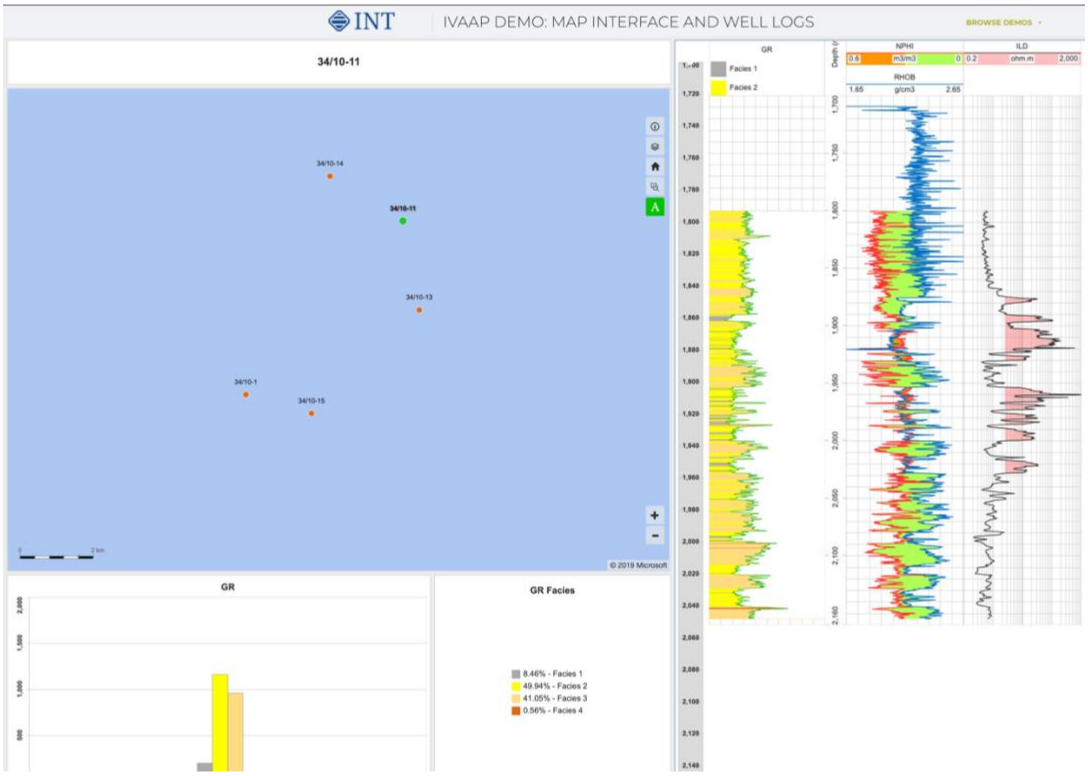


Figure 3.2 IVAAP Demo on Map Interface and Well Logs [77]

3.2.3 Schlumberger's ProSource Front Office

Schlumberger was built in 1926 with a vision to provide energy industries with richer and deeper data insights [78]. Schlumberger has provided many products throughout the time, including those mentioned earlier ProSource Front Office. The ProSource Front Office is compared with the author and team's VDR as it has an integrated GIS feature similar to the author's GIS. The GIS of ProSoruce Front Office includes visualizing oil and gas fields, wells, seismic lines, and seismic surveys in a location on a map [14]. Aside from that, the components on the map include map toolbox buttons, a map navigator, and a map right-click menu [14]. The users of ProSource Front Office can also pan, zoom in, and zoom out the map [14]. As shown in Figure 3.3, the map window contains several components represented by numbers.

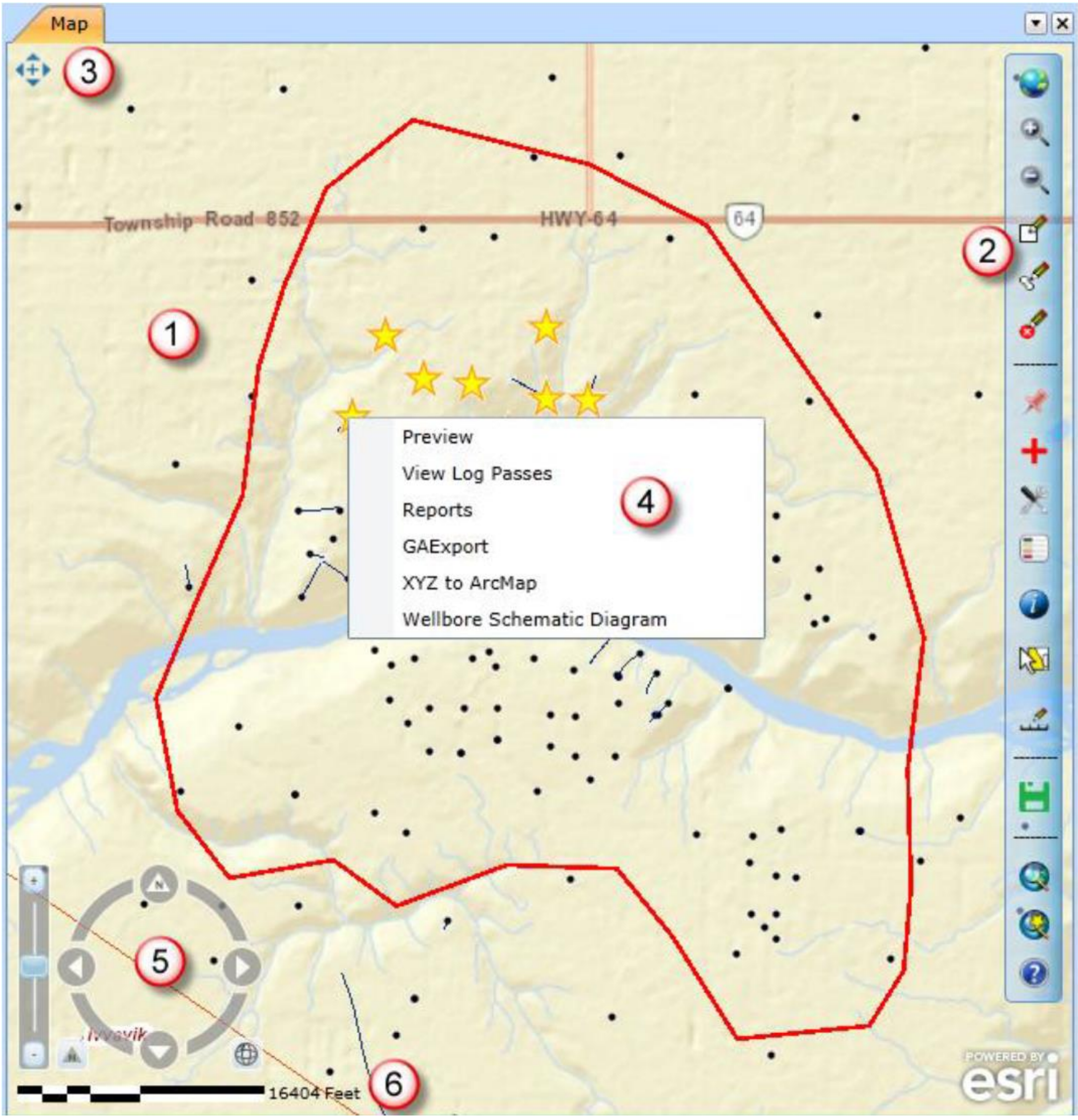


Figure 3.3 Schlumberger ProSource Front Office's Map Window [14]

Number one in Figure 3.3 shows the graphical view of the fields, wells, seismic lines, seismic surveys, and leases [14]. Number two is the map toolbox buttons [14]. Each button has different functionality. For instance, a button is used to zoom in, another button is used to zoom out, another button is used to save the map display, and many others [14]. Number three is the open tool button which allows users to open the previously accessed toolbox map menu [14]. Number four represents the map right-click menu where users can choose an action based on the selected object on the map

[14]. Number five is the map navigator, which allows the user to move around the map and alter the scale [14]. Number six is the scale bar that shows the current scale [14].

3.3 Proposed Solutions

As mentioned in Section 1.2.1, the author and team decided to develop the VDR website application with OSS and is enable for future customization. There many features related to GIS that exist in existing applications which is mentioned in Section 3.2. However, some of the features are less necessary according to the client. Therefore, the author is responsible to develop the only required features but not restrained that the features are similar with existing application.

As the author implemented GIS in the map visualization feature and map data showcase feature of VDR website application. The map visualization feature is similar to Lynx Virtual Data Room and Schlumberger's ProSource Front Office where a map will display data based on the location of the data. The other similarity of VDR Website application and Lynx Virtual Data Room include the ability for user to choose the style of the displayed data, maximize and minimize the display card of the data, zoom in and out of the display card of the data. The differs between both applications are the data provided and the display style that the user can choose. The data provided by Lynx Virtual Data Room are 2D seismic lines, 3D seismic, Wireline logs, and Well reports/seismic processing reports. While the data provided by VDR Website application are 3D seismic line, 3D well, and 3D surface. The different on the display style is the Lynx Virtual Data Room only provide 5 styles to be chosen. While the VDR Website Application provide more than 5 styles.

Furthermore, the data showcase feature is inspired from INT's IVAAP where there are tooltips on the map that indicate the location of the wells. The tooltips are clickable. When the tooltips clicked, the information and images related to the well selected will be display. The tooltips can also be filter based on the well. The information of the well selected can also be download into pdf for the user to keep. Other than these abilities, there is other ability that the author add to the feature based on the requirement of the product owner and client. The ability mentioned is the ability to sort the tooltips on the map based on oil and gas volume.

Based on the explanation above, the differences between the proposed solution and existing application can be summarize as shown in Table 3.1.

Table 3.1 Differences between the Proposed Solution and Existing Application

Function of the features	Lynx's VDR	Interactive Network Technologies (INT) 's IVAAP	Schlumberger's ProSource Front Office	Proposed Solution for the GIS feature inside the VDR Web app
Filter the Map based on the oil and gas fields/wells	Not Available	Available	Not Available	Available
Sort the Map based oil and gas volume	Not Available	Not Available	Not Available	Available
Overlay	Available	Not Available	Not Available	Available
Data visualization/analysis such as seismic line, well log, and surface can be chosen based on oil and gas location	Available	Available	Available	Available
Oil and Gas well data showcase can be chosen based on oil and gas location	Not Available	Not Available	Not Available	Not Available
The style of the data visualization can be changeable	Available	Not Available	Not Available	Available

With the features decided to be develop using open-source frameworks and libraries, the author is left with many options in choosing the frameworks and libraries. However, the author decided to use Vue Js, Material Design, Vuetify, and Leaflet.

3.3.1 Front-End Framework Used

The author decided to use Vue Js due to certain advantages more than React Js. The advantages mentioned in Section 2.2.1 include the optimization efforts and beginner-friendly templates Vue Js gave. As the author had previous experience with HTML, the author can quickly adapt to Vue templates. The author can also easily visualize all components in a single file, including CSS.

3.3.3 Front-End Libraries Used

In order to enhance the Front-End, some libraries are used such as vue-draggable-resizable and vue-router. The vue-draggable-resizable library allow the element in the website to be easily draggable and resizable and the author can easily implement the library as a Vue component. In other hand, the vue-router is the official router for Vue.Js to navigate through components.

3.3.2 Front-End Design Framework Used

To support the front-end, the author chooses Front-End design framework that supports Vue Js, which is Vuetify, that is supported by Material Design, as explained in Section 2.2.3. Although many other Vue UI/UX frameworks exist, Vuetify has the most thorough features shown in Figure 2.4, which beat BootStrapVue, Buefy, Element UI, and Quasar. On another side, Material Design is chosen over other design systems due to some prestige, such as extensive documentation, providing flexibility and freedom for the designers to implement the design, and components provided assemble how it works in the real world but simplified [79]. Material Design is also chosen because it covers the trends in web designs; skeuomorphism, and flat design [80]. It mostly combines flat design elements with skeuomorphism’s 3D touches, which means more interaction and great user experiences [80]. Therefore, many modern applications are using Material design, such as Lyft, Telegram, and BuzzFeed [81]. Lyft is a ride-hailing application that allows users to book a car or bike [81]. Lyft uses Material Design to make the application more interactive and provide a better experience for users, which is proven successful as it was a top-rated Android app in 2021 [81]. As Lyft also integrated GIS, the author is inspired to design the GIS inside the VDR using material design [81].

3.3.3 GIS Framework Used

As the author is responsible for integrating GIS into the VDR website application, the author has chosen Leaflet as the GIS library. The author chose Leaflet due to Leaflet's superiority over other GIS libraries such as Open Layers, which is important as the thesis is only given a short time for development. The Leaflet's superiority over Open Layers is shown in Table 3.2.

Table 3.2 Leaflet vs OpenLayers [82]

	Leaflet	OpenLayers
Learning Curve	Shallow (Beginner-friendly)	Steep
Documentation	More structured and complete with a bunch of examples and tutorials	Extensive but outdated
Community	Larger community	Smaller community
Amount of Code	Minimalist code when implementing	More code when implementing

Moreover, the author uses Vue2Leaflet and vue2-leaflet-geosearch which is mentioned in Section 2.3.2. Vue2Leaflet is used in order to integrate Leaflet with Vue Js easily. With Vue2Leaflet, the author can just use Vue components to construct the map. While the author uses vue2-leaflet-geosearch to easily utilize Vue components to embed geocoding (address lookup) to the Leaflet map.

CHAPTER 4
SOLUTION DESIGN

This chapter explains and displays the design and the flow of the GIS application in the proposed VDR web application.

4.1 System Architecture

The system architecture of the website application is illustrated in Figure 4.22. The users shall be able to access the 6 main components which are the main features of the website application based on the user types. The user types are premium, regular, and administrator. The author is responsible in developing the frontend for two of the features as shown in Figure 4.21, which are the:

- Data showcase page, and
- Map page

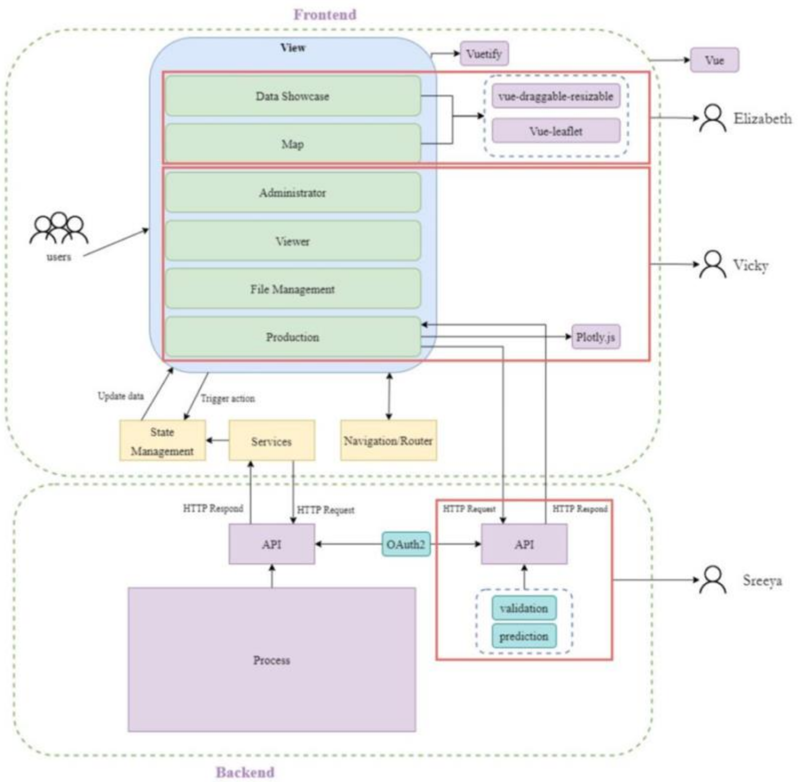


Figure 4.1 System Architecture

The frontend development is develop using Vue as the frontend framework and Vuetify as the frontend design framework which are mentioned in Section 3.3.2 and Section 3.3.3. Other than that, two third-party Vue libraries, vue-draggable-resizable and Vue leaflet are used for the features that the author responsible of. Furthermore, the author used vue-router for the navigation router. The *Services* will be handling all the request and response to and from the APIs. The API endpoints is provided by the product owner. While the backend is provided by the third party. Aside from that, the author and team follow Scrum methodology in developing the application so development is done in sprints and documented in Jira. The Jira sprints are as shown in Figure 4.2.

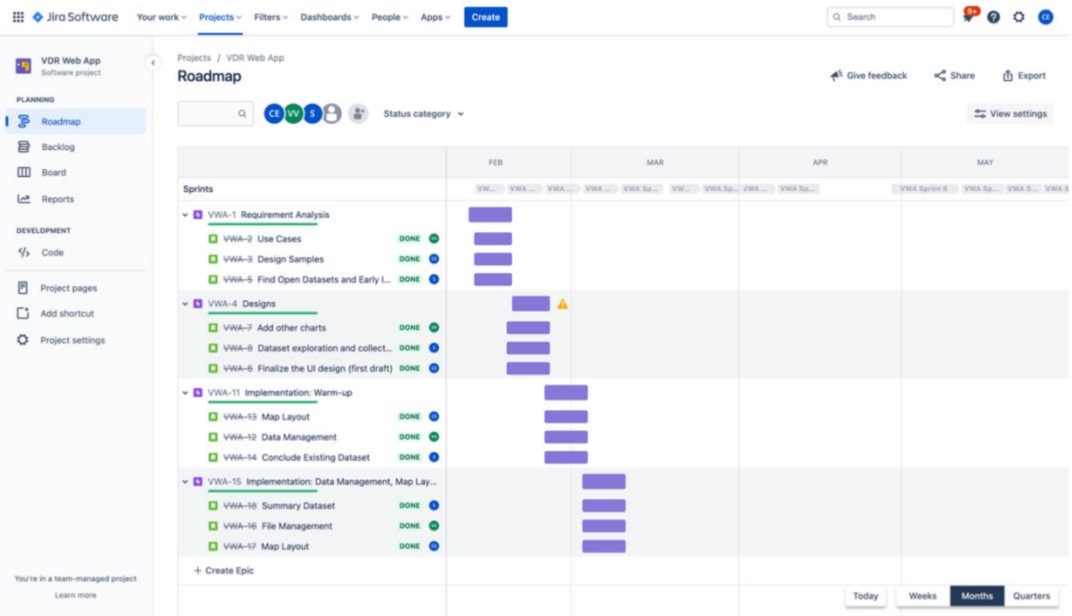


Figure 4.2 Project Roadmap in Jira

4.2 Sitemap

Sitemap for users is illustrated in Figure 4.2.

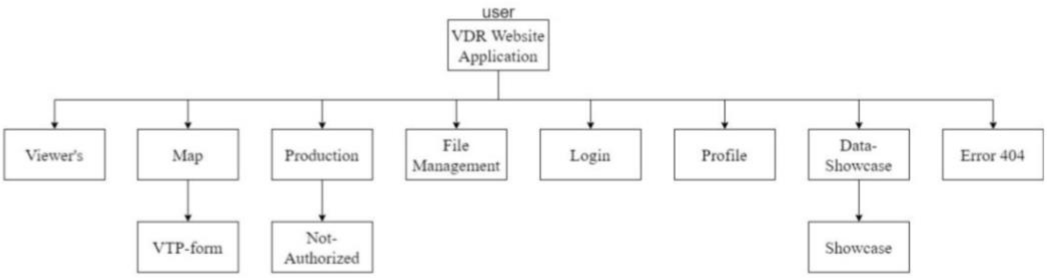


Figure 4.3 Sitemap

4.3 Use Case Diagram

For this thesis, the author and team use the Use Case Diagram, one of the Unified Modelling Language (UML) design techniques. The use case diagram is used to help developers to visualize the functionality of the system from the users' point of view, as mentioned in Section 2.6.

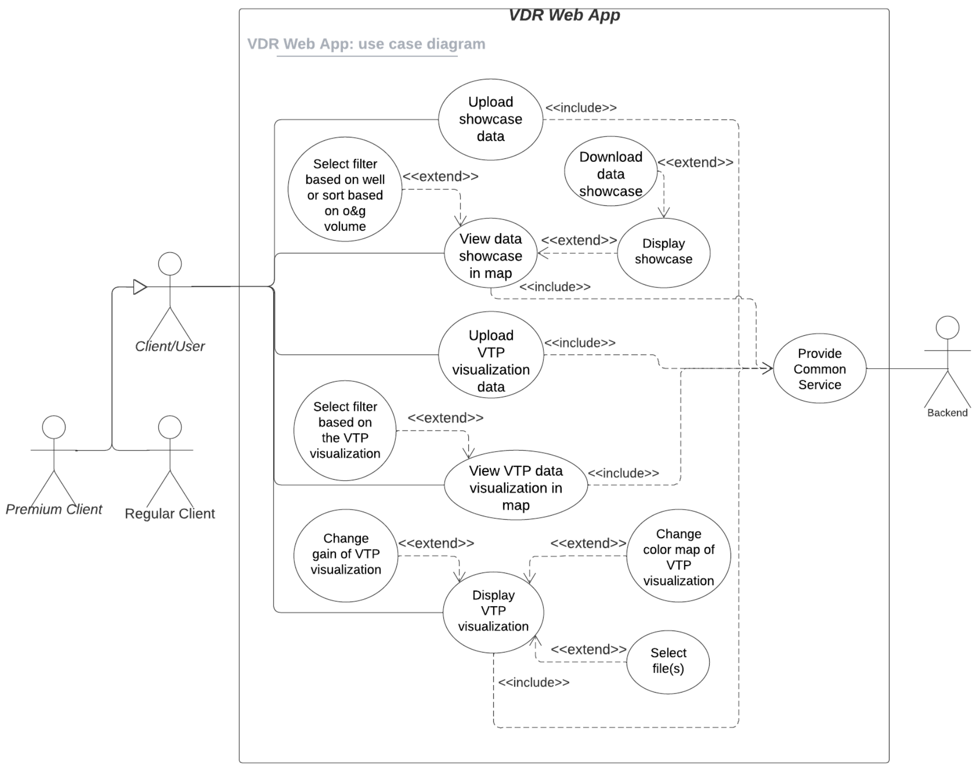


Figure 4.4 Use case diagram of the Client (Map Function)

Figure 4.4 shows the use case diagram for the map functionalities that the author is responsible of. The primary actors in the use case diagram are the premium or regular clients. The rectangle section represents the system, and inside it is the web application's functions that the client can do. There are five functions. Each function is required to connect to the common service because it needs the service to run. The common service is a backend service that contain the API endpoints. This service is provided by the backend. The backend developers are other developers such as geoscientist from Pertamina University.

4.4 Flowchart

Aside from the use case diagram, the author uses a flowchart to illustrate the computer system, algorithm, and process, as mentioned in Section 2.6.

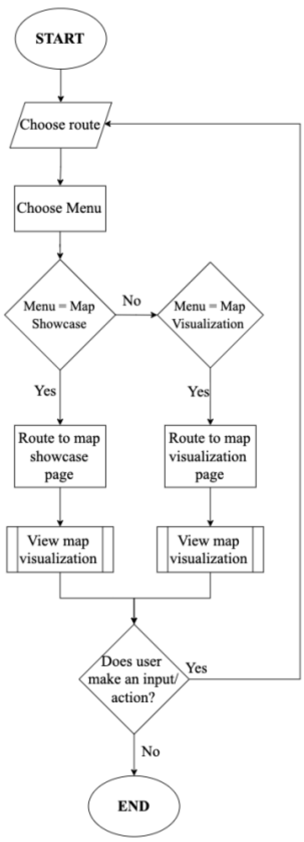


Figure 4.5 Flowchart of Map Showcase and Map Visualization

Figure A.1 in the Appendix shows the general flowchart of the processes that the clients/users can perform. The processes that the author is responsible for developing are shown in Figure 4.5. In Figure 4.5, a rectangle with vertical lines on the right and left sides represents the predefined processor which is presented in Figure 4.6 and Figure 4.7. Inside the predefine processors, there are another predefined processors. The predefined processor in Figure 4.6 is presented in Figure 4.8. While the predefined processor in Figure 4.7 is presented in Figure 4.9.

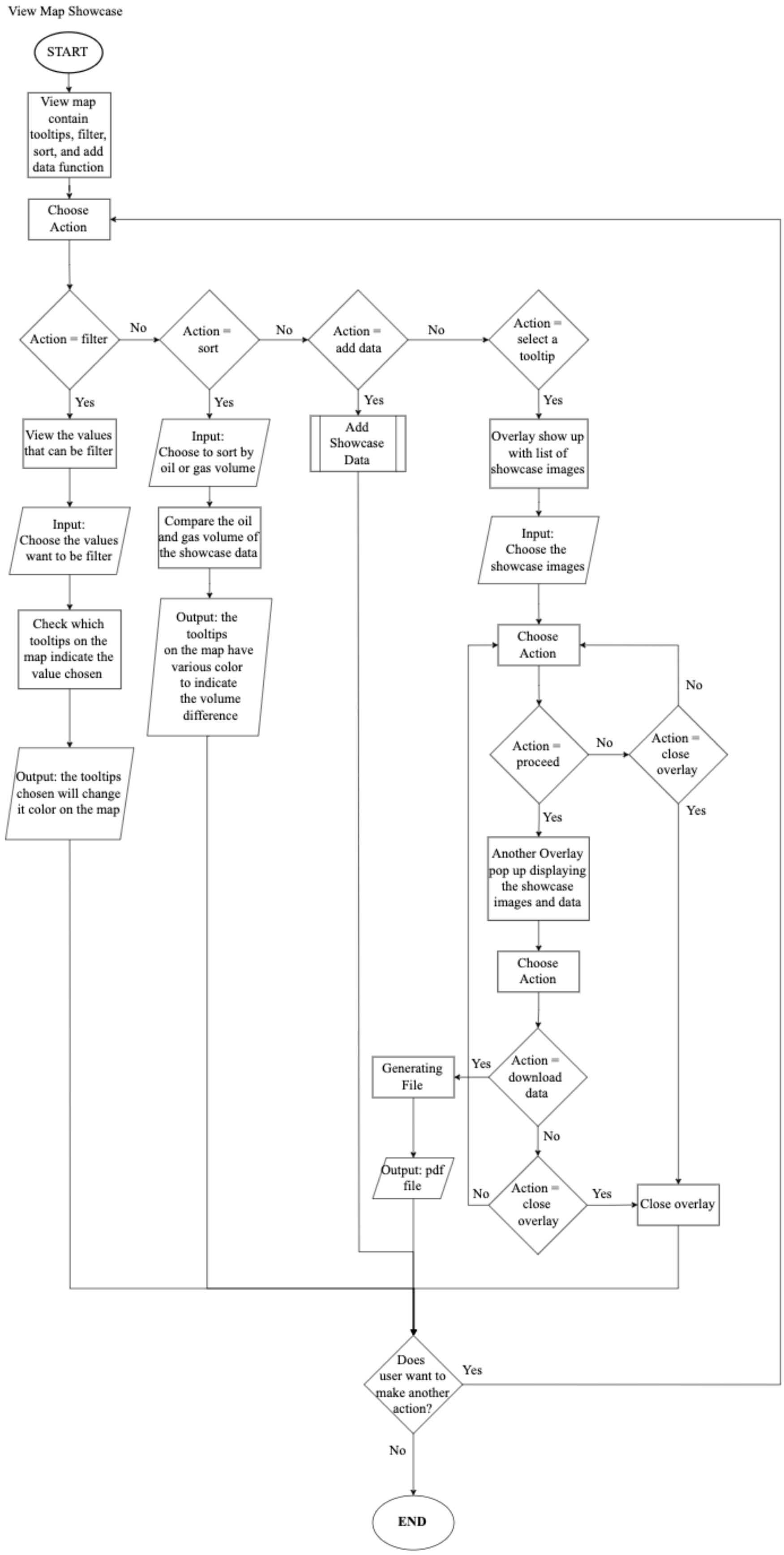


Figure 4.6 Predefined Processor View Map Showcase

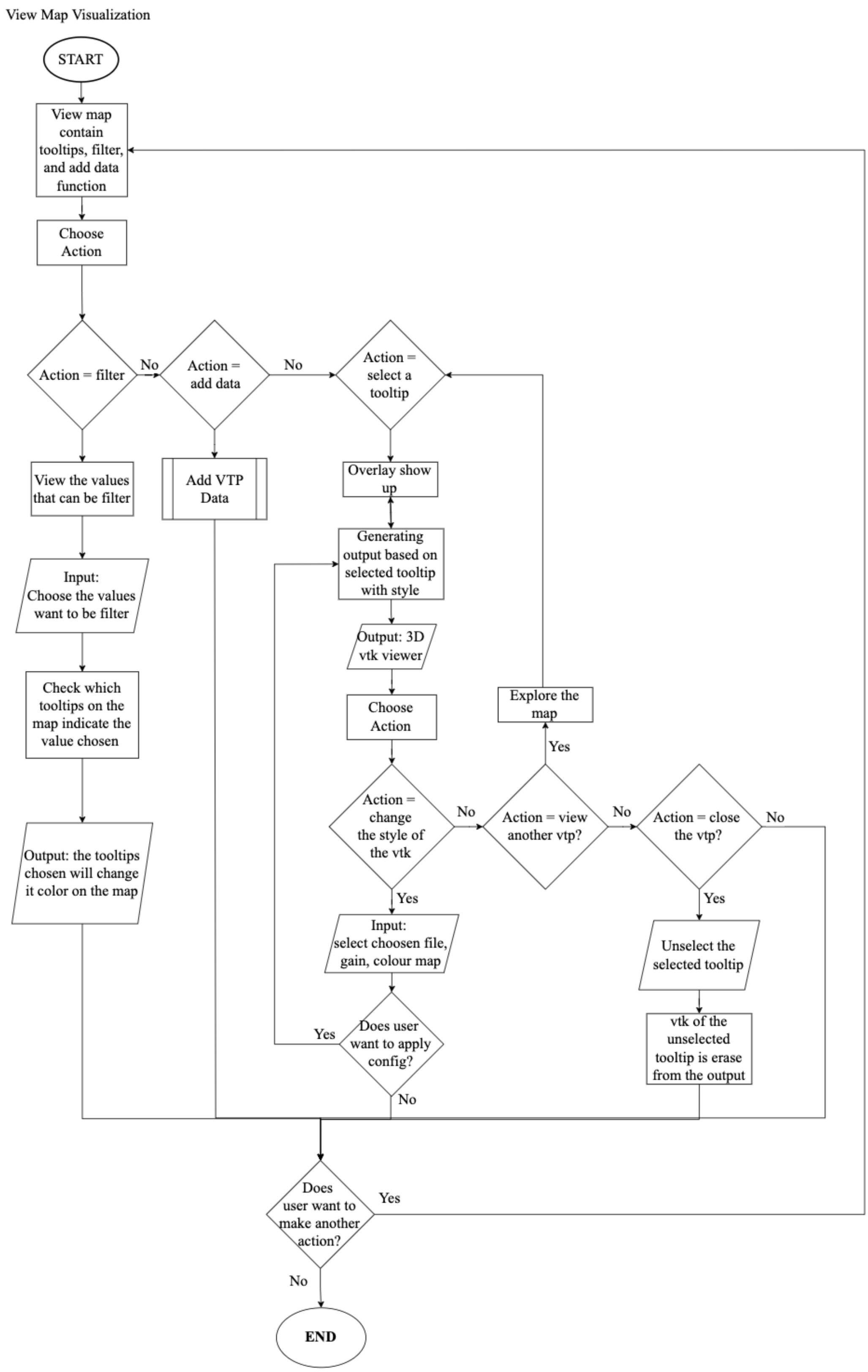


Figure 4.7 Predefined Processor View Map Visualization

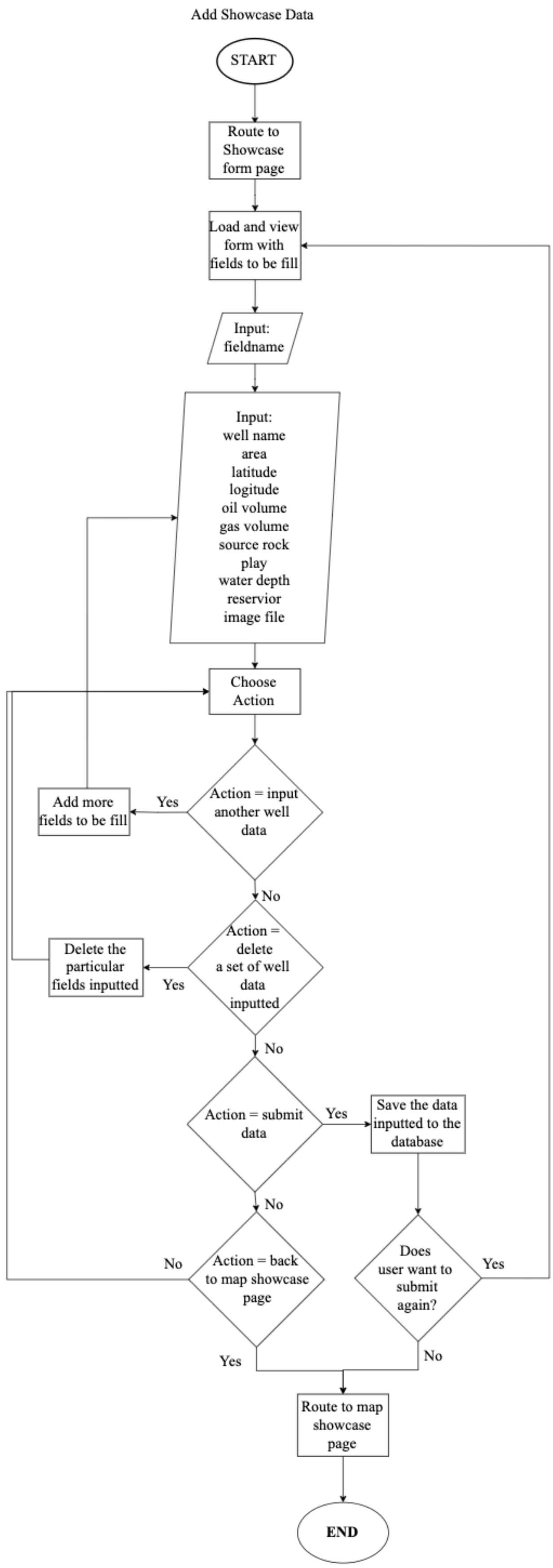


Figure 4.8 Predefined Processor - Add Showcase Data

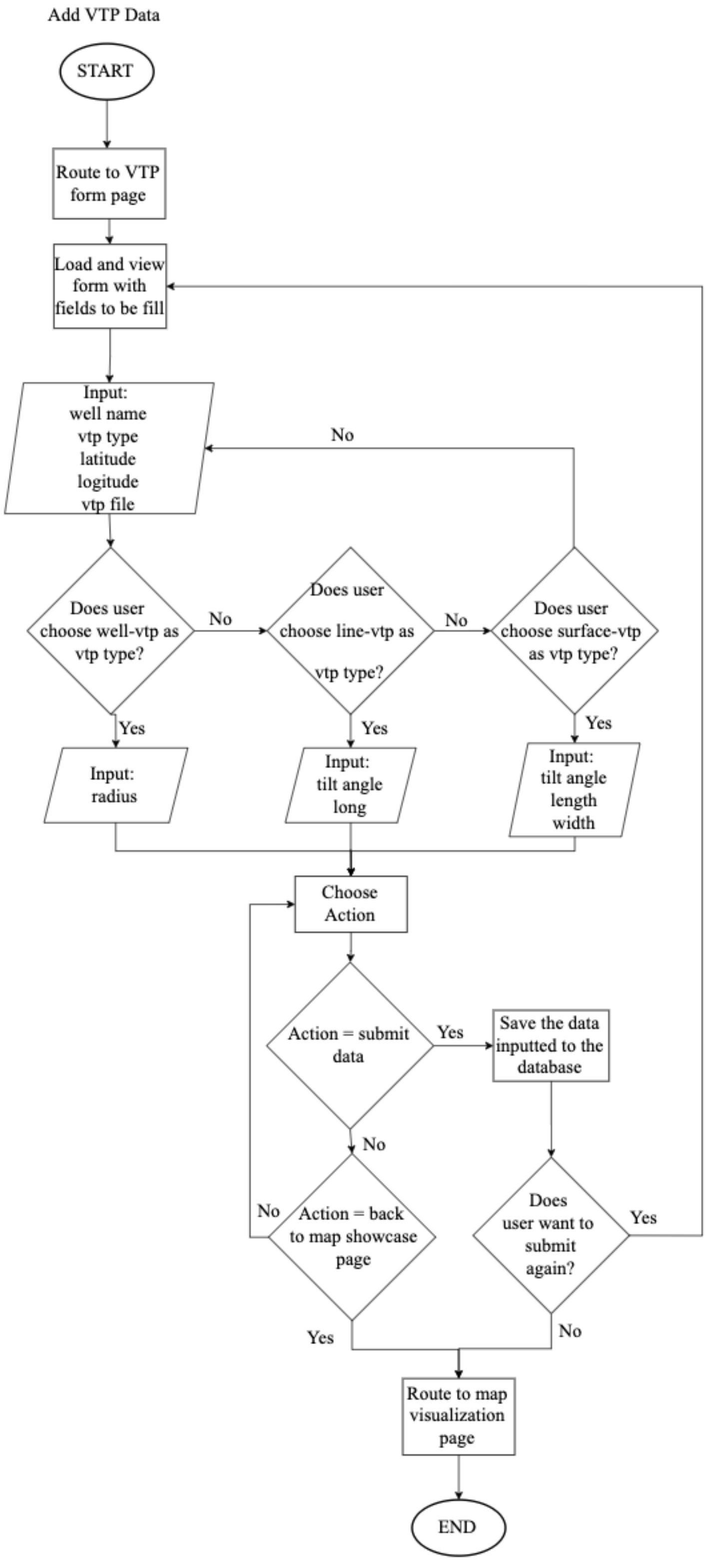


Figure 4.9 Predefined Processor - Add VTP Data

4.5 Wireframe

The initial design in the form of wireframe of the website pages that the author responsible for as mentioned in the flowchart are shown in Figure 4.8 – 4.14. The wireframes are made according to components and elements provided by Vuetify.

4.5.1 Map Visualization (3D Viewer Map) Page

The design of the map visualization is as illustrated in Figure 4.10. The main component in this page is the map that covers almost a whole page. There is also tooltips above the map in a form of circles, rectangles, and lines that represent well logs, surfaces, and seismic lines. Then, there is also draggable and resizable component wrapping Vuetify tree view above the map which give users the choice to filter the tooltips on the map.

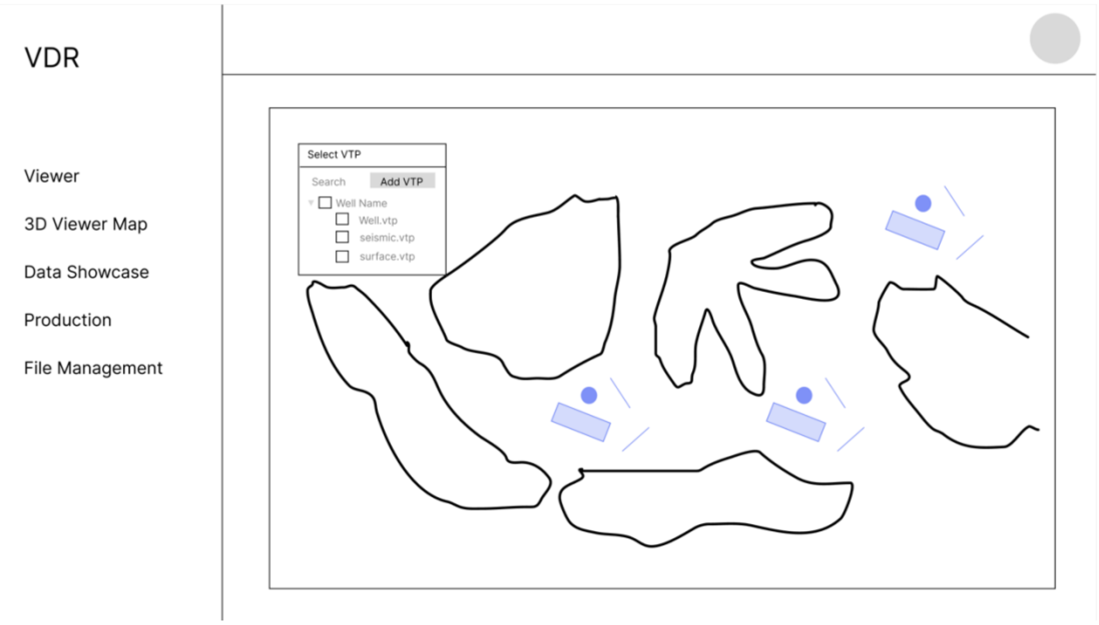


Figure 4.10 Map Visualization (3D Viewer Map) Page - First Onload

When a VTP file from the tree view is selected, the tooltip that represent the particular VTP file turns darker as shown in Figure 4.11.

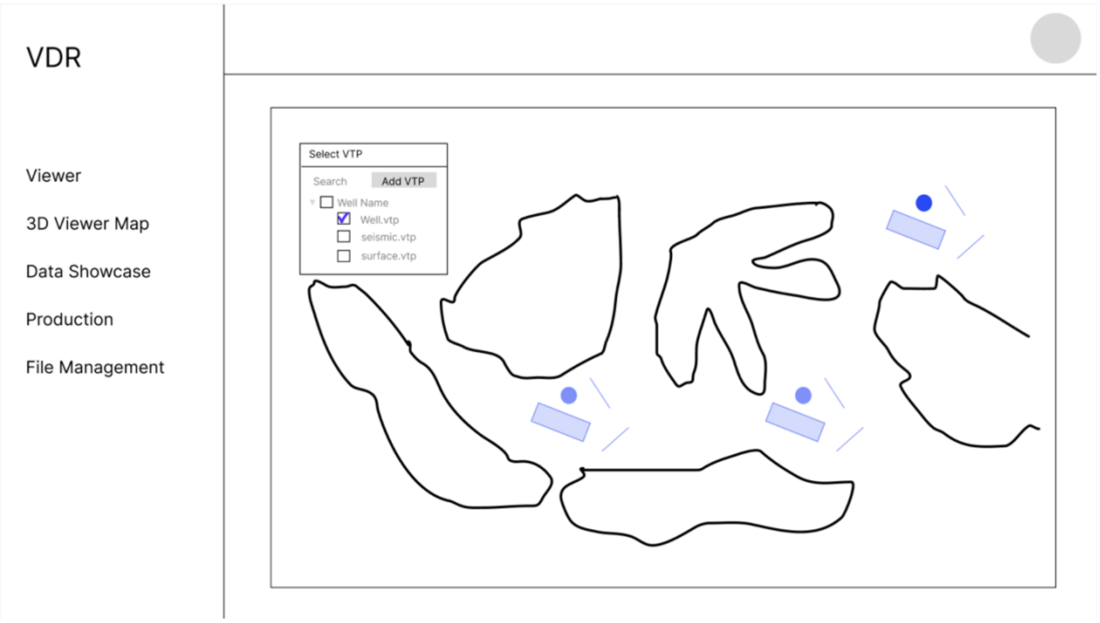


Figure 4.11 Map Visualization (3D Viewer Map) Page - Filter Applied

When a tooltip on the map is selected, two draggable and resizable components will show and the tooltip turn colour as shown in Figure 4.12. The first component wrap three different form input components and a button. The three form input components include 3D select dropdown, Gain Slider, and Colour Map Dropdown.

- 3D select dropdown

  The select is a dropdown which will consist the list of files the user selected based on the tooltips clicked. When tooltips is clicked, the following changes will be applied to the file selected in the column.

- Gain Slider

  The gain is the data which will modify the texture of the data chosen using the 3D select. The gain is in the form of slider which allow users to select a single value by sliding the slider.

- Colour Map Dropdown

  The colour map in form of dropdown allow user to change the colour of the VTP data.

The second draggable and resizable component contain the display of well log, seismic data, or surface as the visualization of the chosen VTP file.

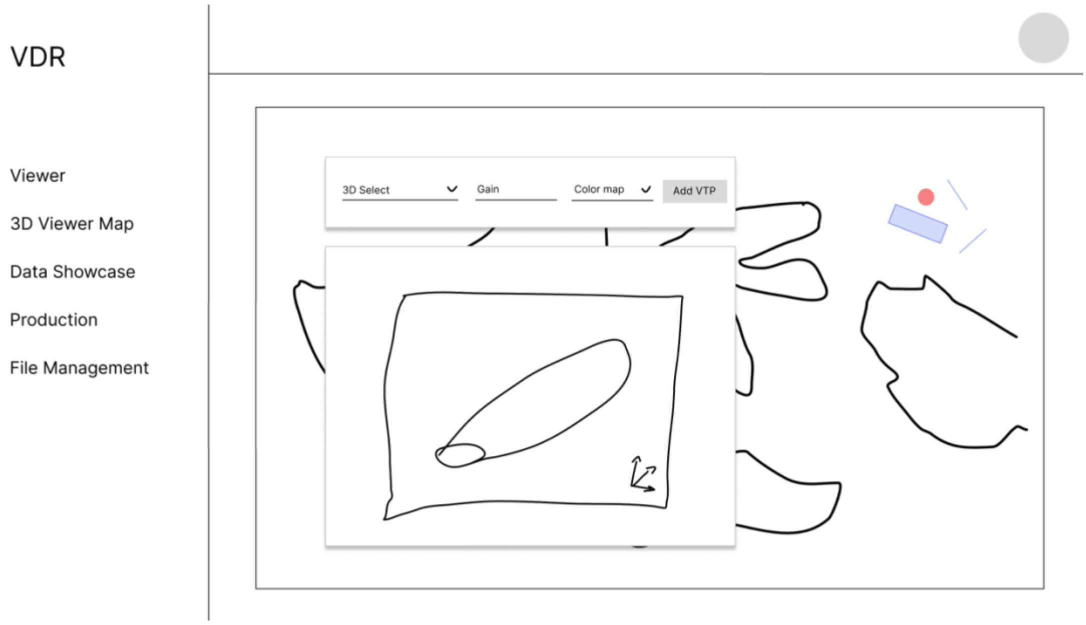


Figure 4.12 Map Visualization (3D Viewer Map) Page - Tooltip Onclick

The button on the first draggable and resizable component allow user to redirect to VTP form page where user can add more VTP data and file.

4.5.2 VTP Form Page

The VTP form page is as shown in Figure 4.13. In this page, almost all the components are the form input components such as dropdown, file input, and text field. The component that is not form input is the map. In this map, the users need to double click the VTP location on the map or search the address of the VTP location. The chosen location will then transform into latitude and longitude to be store. The back button in this page will redirect users back to Map Visualization (3D Viewer Map) Page. While the submit button will store all the data inserted to the fields of the form into the database.

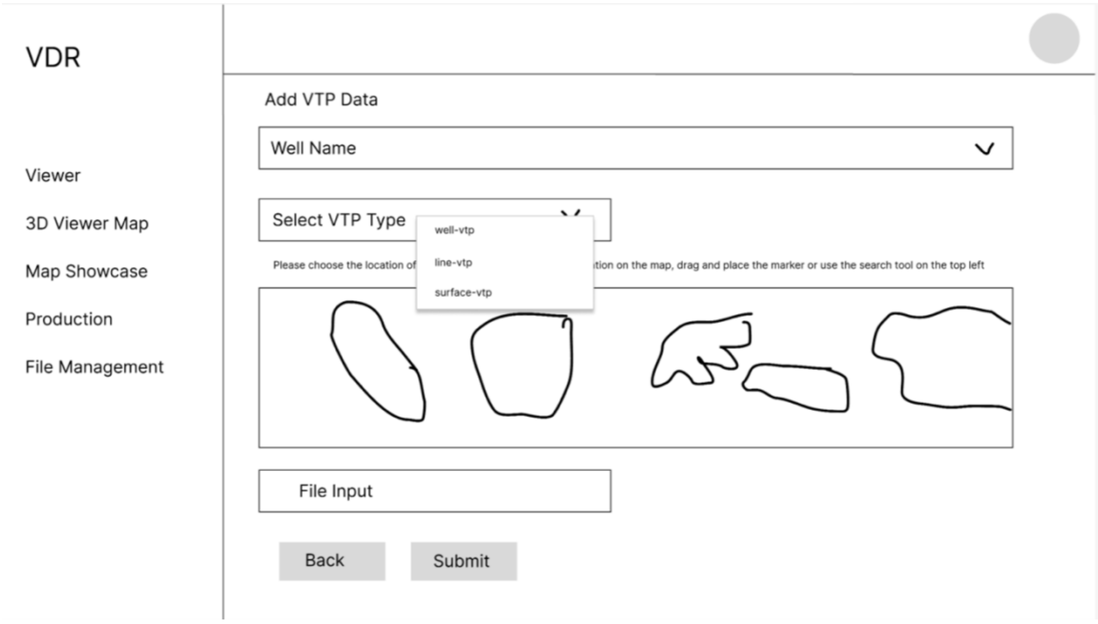


Figure 4.13 VTP Form page

In this page, there are also different form inputs that show up based on the VTP type selected. When *well-vtp* selected as the VTP type, the form with label *Radius* will show and requires the users to fill the particular form as illustrated in Figure 4.14. When *line-vtp* selected as the VTP type, the form with label *Tilt Angle* and *Long* will show and requires the users to fill the particular form as illustrated in Figure 4.15. When *surface-vtp* selected as the VTP type, the form with label *Tilt Angle, Length,* and *Width* will show and requires the users to fill the particular form as illustrated in Figure 4.16.

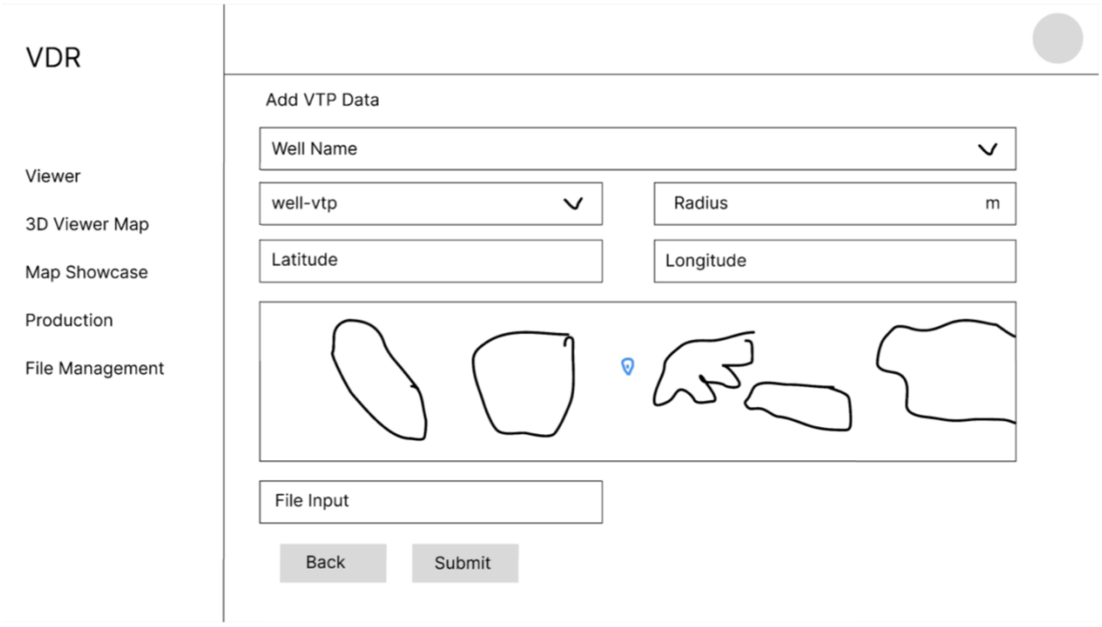


Figure 4.14 VTP Form Page - Selected well-vtp as VTP type

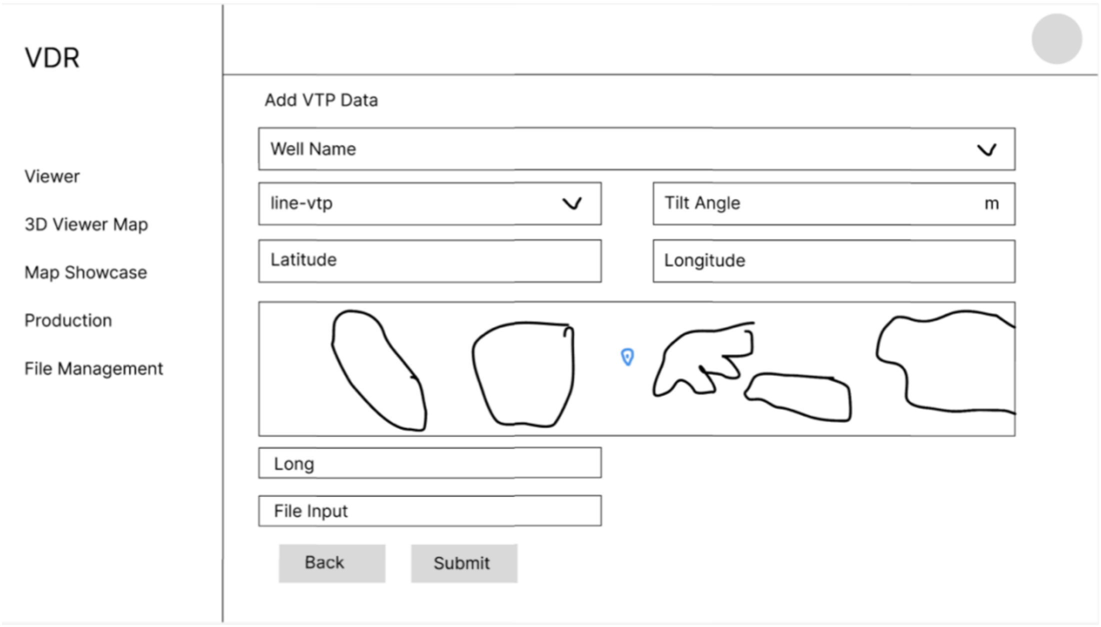


Figure 4.15 VTP Form Page - Selected line-vtp as VTP type

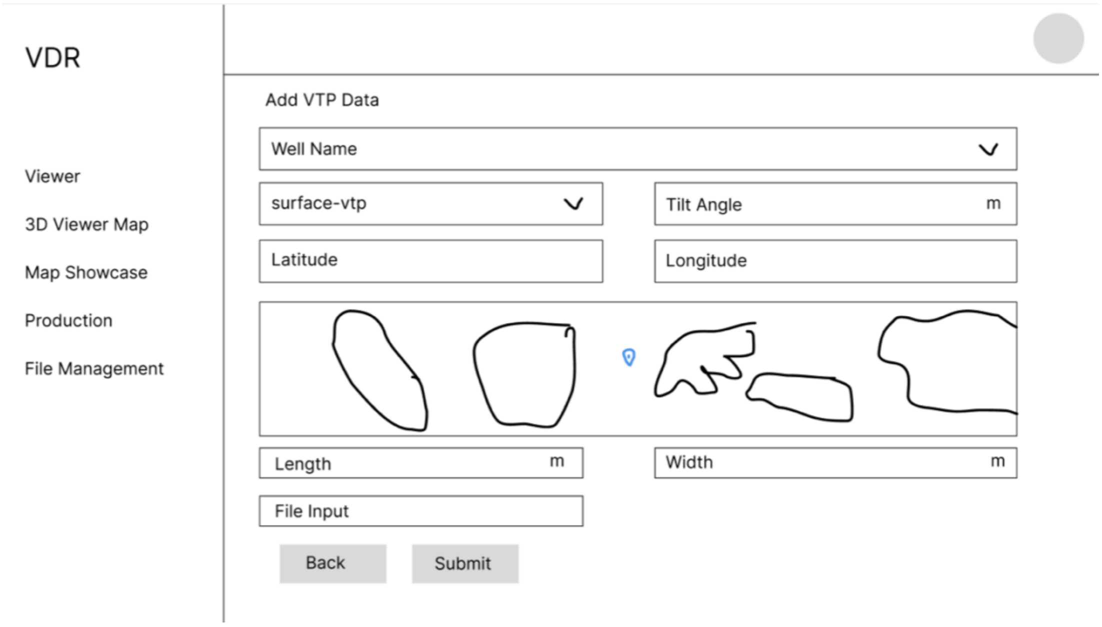


Figure 4.16 VTP Form Page - Selected surface-vtp as VTP type

4.5.3 Map Data Showcase Page

The map data showcase page design is as illustrated in Figure 4.17. The design is similar with the design of Map Visualization (3D Viewer Map) Page. The main component in this page is the map that covers almost a whole page. There is also tooltips above the map in a form of circles that represent the well. Then, there are two draggable and resizable components and draggable button. First draggable and resizable component wrapped Vuetify Treeview above the map which give users the choice to filter the tooltips on the map. Second draggable and resizable component wrapped Vuetify dropdown above the map which give users the choice to sort the tooltips on the map based on oil volume and gas volume as shown in Figure 4.17. The draggable button in this page allow users to add more data to the map for showcase by redirecting the user to Data Showcase form page.

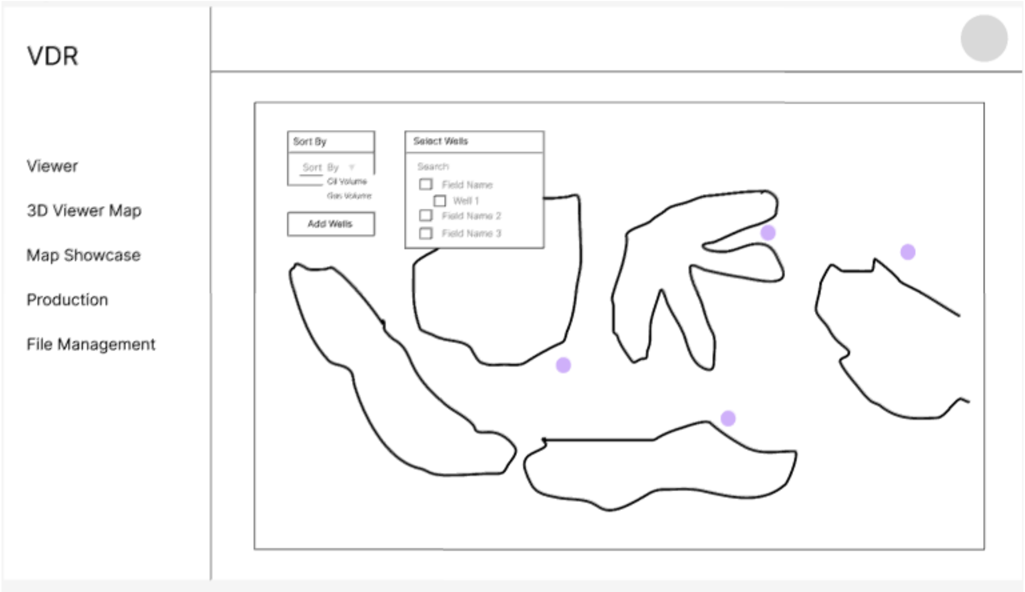


Figure 4.17 Map Data Showcase Page

When well is chosen using the filter tree view, the tooltip of the well will change its colour as shown in Figure 4.18.

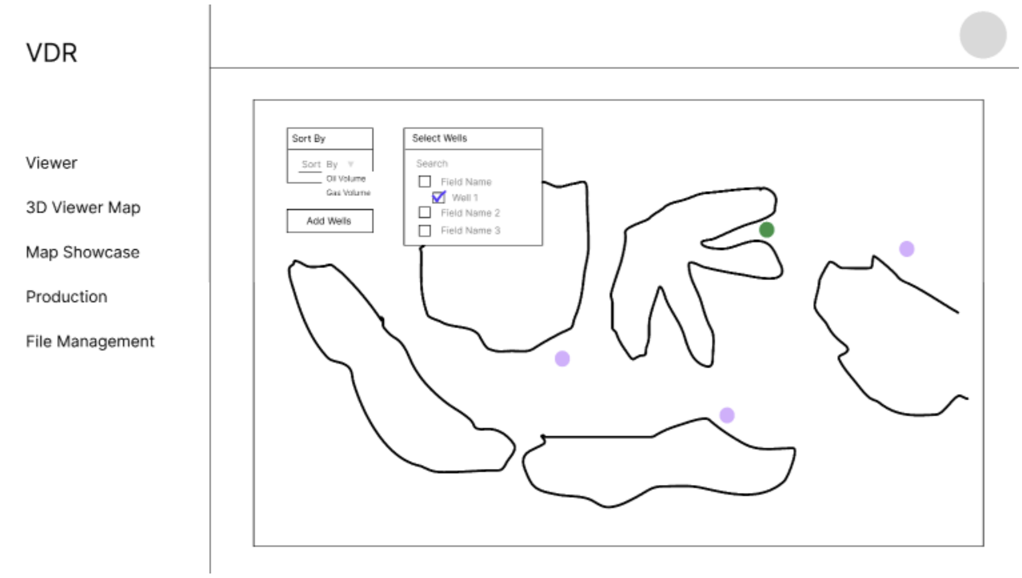


Figure 4.18 Map Data Showcase Page - Filter Applied

When user choose to sort by oil volume or gas volume, the outer circle will show range of colours as shown in Figure 4.19. The darker colour indicate more volume. While the lighter colour indicate less volume.

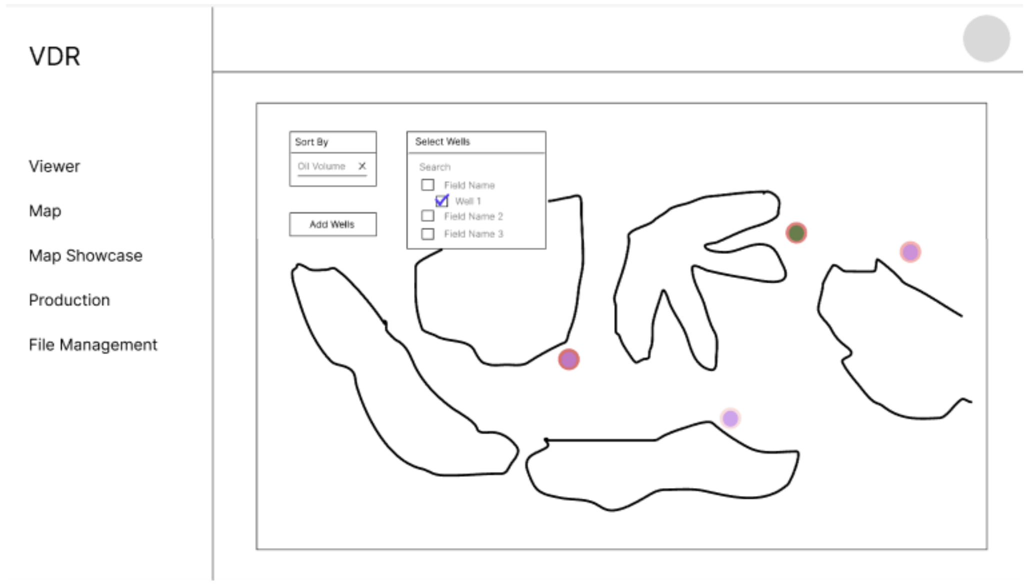


Figure 4.19 Map Data Showcase Page - Sort & Filter Applied

When the tooltip is click, an overlay will show that allow users to choose the checkboxes of showcase images of the particular well that the user want visualize as shown in Figure 4.20. When users do not choose any of the checkboxes, the users cannot proceed as the button is disable and can only close the overlay.

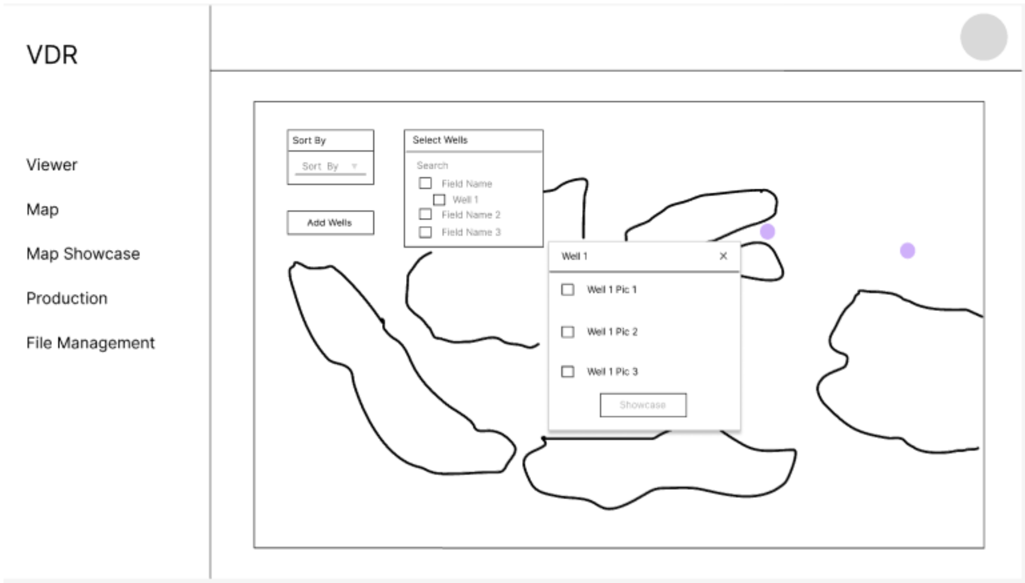


Figure 4.20 Map Data Showcase Page - Tooltip Onclick

When one or more checkboxes selected, the proceed button is clickable as illustrated in Figure 4.21.

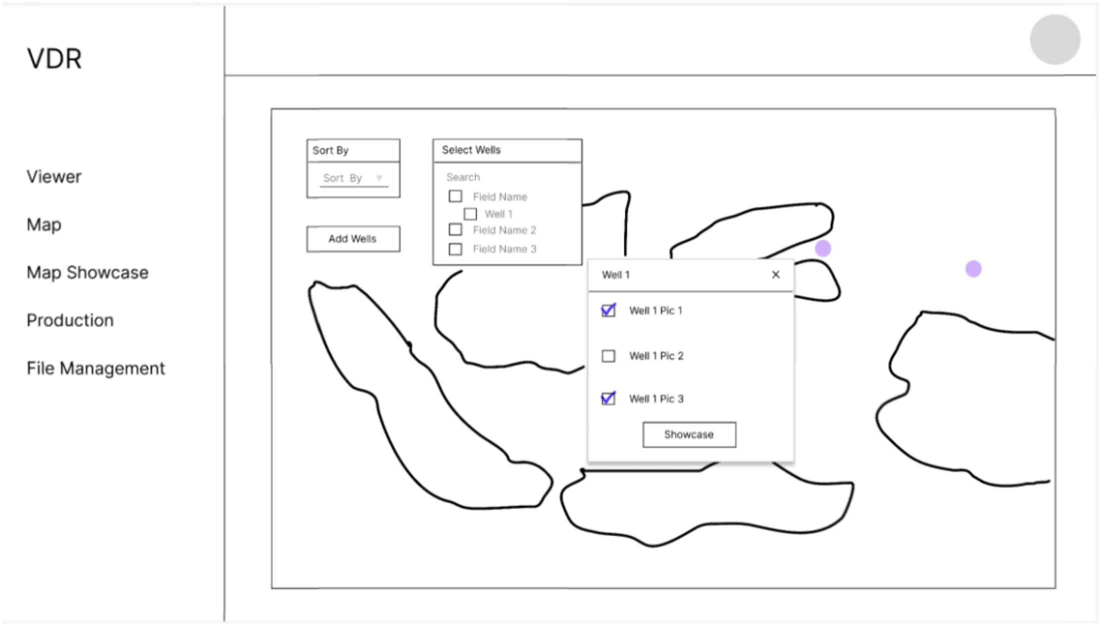


Figure 4.21 Map Data Showcase Page - Selected one or more checkboxes

When the proceed button is click, another overlay will show containing the showcase images previously chosen by the users and the showcase data as illustrated in Figure 4.22. Aside the showcase images and showcase data, there are a two icon buttons. The first icon button is download button which is used to download the image and data in form of a pdf. The second icon button is close button which has the functionality to close the overlay.

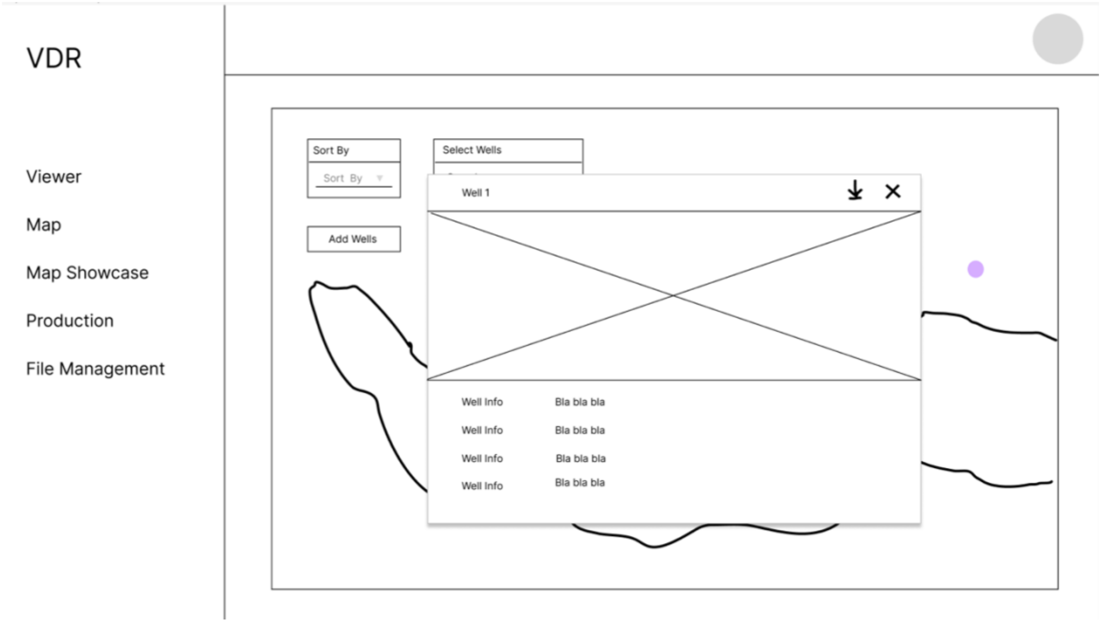


Figure 4.22 Map Data Showcase Page - Data Showcase Overlay

4.5.4 Data Showcase Form Page

Data Showcase Form Page is the landing page after the draggable button in Map Data Showcase Page is clicked. The main component of this page is the form where users are require to fill in several fields regarding the showcase data as shown in Figure 4.23. The button 'Add Wells' in this page will add more fields to fill more wells. While the button 'Delete' will delete a set of fields. The back button in this page will redirect users back to Map Data Showcase Page. While the submit button will store all the data inserted to the fields of the form into the database.

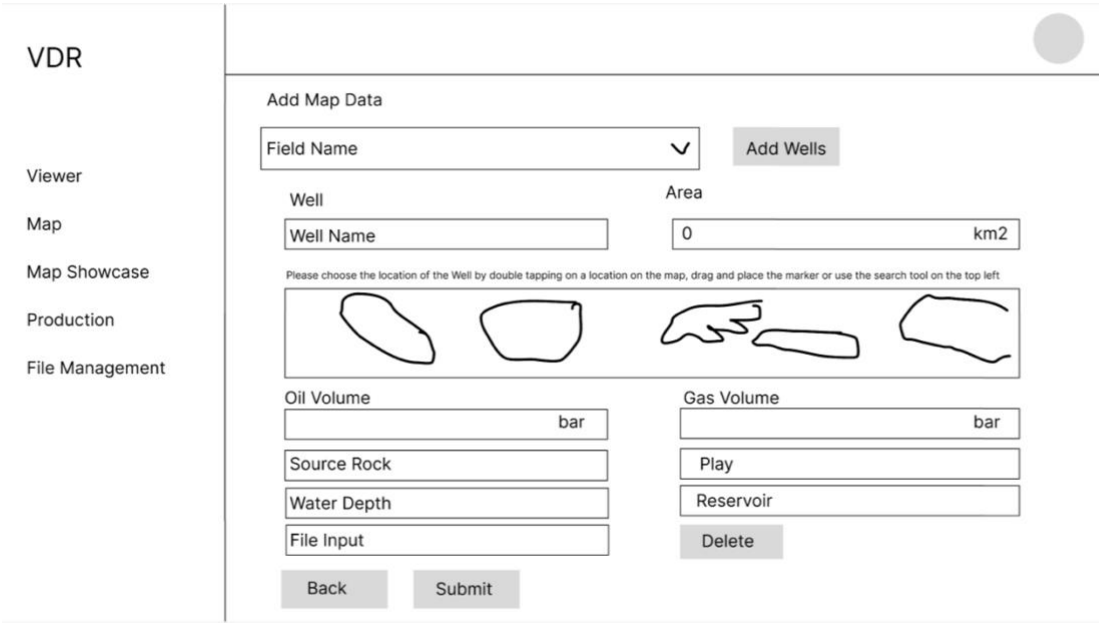


Figure 4.23 Data Showcase Form Page

When the users double tap on a point on the map in this page, the latitude and longitude filled fields will show with the latitude and longitude of the point as illustrate in Figure 4.24.

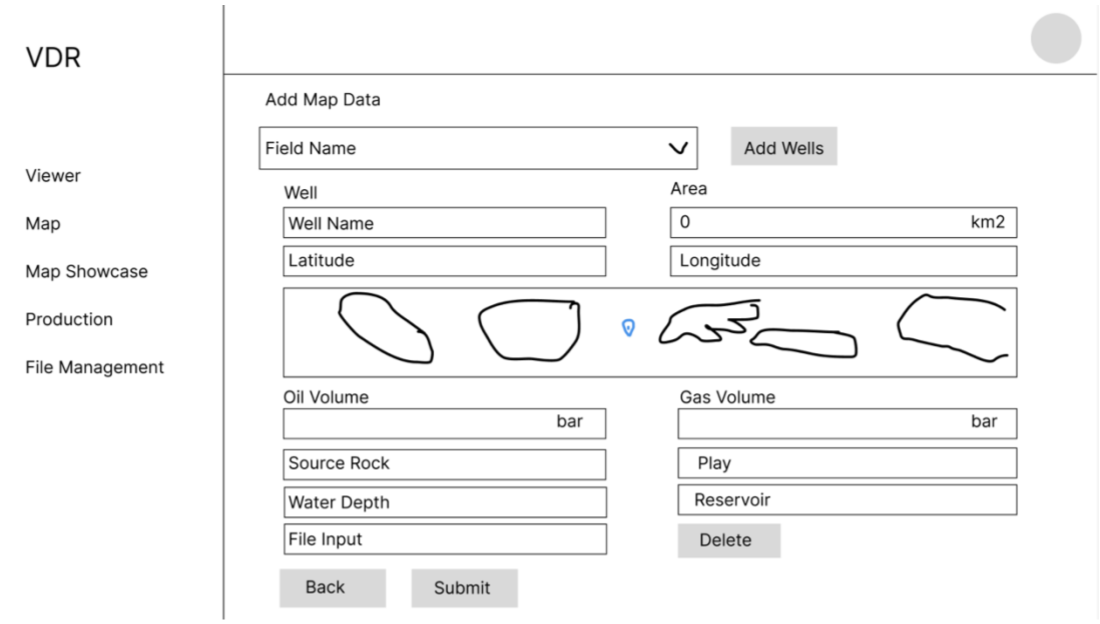


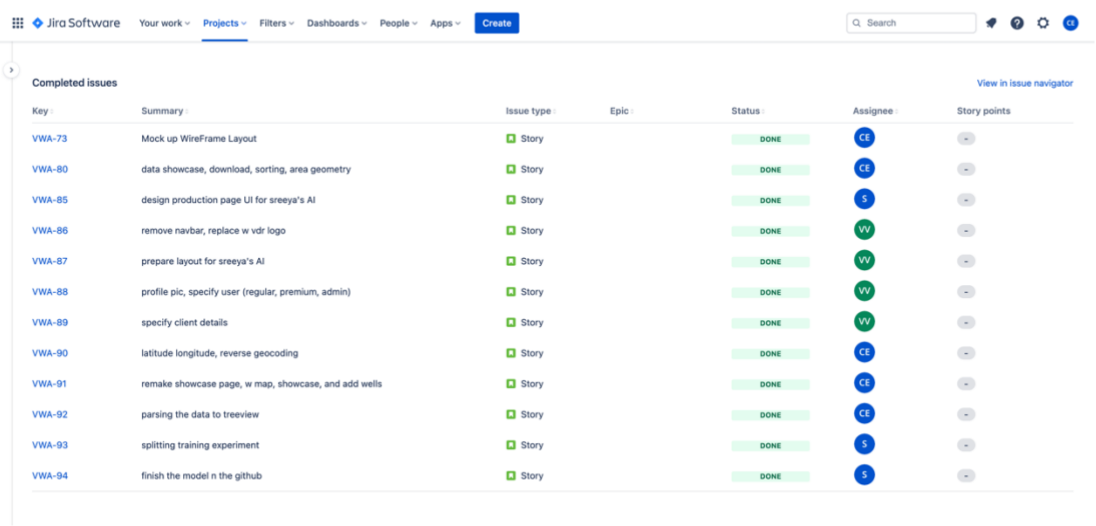
Figure 4.24 Data Showcase Form Page - Double click a point on the map

CHAPTER 5
IMPLEMENTATION

This chapter explains the cycle of developing the application and the route of the web application to the pages that the author is responsible for. Then, the endpoints to communicate between the website and backend. Last, the final user interface developed based on the wireframe.

5.1 Jira Sprints

As mentioned in Section 4.1, the development of the application is done in sprints. Specifically 13 sprints which take place from 10 February 2022 to 05 July 2022. Each sprint take place for one week and the author and team are given several tasks that need to be complete within the week. By the end of each sprint, the author and team are require to demonstrate their progress to product owner. After reporting the progress, the author and team will be given another tasks for the next sprint. The example of a sprint contain several tasks is shown in Figure 5.1.



Key	Summary	Issue type	Epic	Status	Assignee	Story points
VWA-73	Mock up WireFrame Layout	Story		DONE	CE	-
VWA-80	data showcase, download, sorting, area geometry	Story		DONE	CE	-
VWA-85	design production page UI for sreeya's AI	Story		DONE	S	-
VWA-86	remove navbar, replace w vdr logo	Story		DONE	VV	-
VWA-87	prepare layout for sreeya's AI	Story		DONE	VV	-
VWA-88	profile pic, specify user (regular, premium, admin)	Story		DONE	VV	-
VWA-89	specify client details	Story		DONE	VV	-
VWA-90	latitude longitude, reverse geocoding	Story		DONE	CE	-
VWA-91	remake showcase page, w map, showcase, and add wells	Story		DONE	CE	-
VWA-92	parsing the data to treeview	Story		DONE	CE	-
VWA-93	splitting training experiment	Story		DONE	S	-
VWA-94	finish the model n the github	Story		DONE	S	-

Figure 5.1 Example of Team Sprint

Throughout the sprints, the development progress is moderately stable as the bands shown in the CFD are generally progressing parallelly as depicted in Figure 5.2.

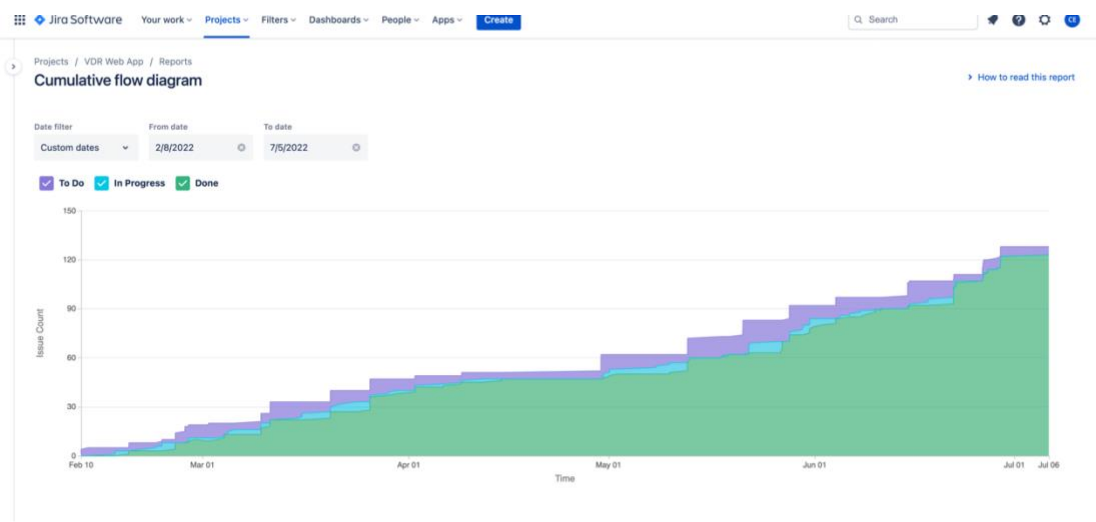


Figure 5.2 Cumulative Flow Diagram

5.2 Route

As mentioned also in Section 4.1, router in the VDR website application is managed by vue-router. With vue-router, the components are managed into routes which ease user to access the components. The routes and components the author is responsible of are as described in Table 5.1.

Table 5.1 Routers

Permission	**Routes**	**Components**
Regular User	/maps	@/views/map/Maps.vue
Regular User	/vtp-form	@/views/map/VTPFormLayout.vue
Regular User	/map-showcase	@/views/showcase/MapShowcase.vue
Regular User	/showcase	@/views/showcase/FormLayouts.vue

All the routers described in the table are only accessible after user's login.

5.3 Communication with API Endpoints

With the API endpoints provided by the product owner as mentioned in Section 4.1. The communication between the frontend with the API endpoints is done using axios. All features that the author responsible of are requires to communicatee with the API Endpoints. As mentioned in Section 3.3, the features are map data showcase and map visualization (3D Viewer Map). The API endpoints for the features are described in the Table 5.2. All the response received from the endpoints are all in the form of JSON. The list of error codes used to validate the endpoints are described in the Table 5.3.

Table 5.2 API Endpoints with Success Response

Endpoints	**Methods**	**Description**	**Request**	**Response (200)**
Map Data Showcase - /api/v1/common				
/showcase	POST	Upload Showcase Data	{"fieldName": string, "wells": [{'name': string, 'area': float, 'latitude':float, 'longitude':float, 'oilVolume':float, 'gasVolume':float, 'waterDepth':"Onshore (darat)", "Offshore shallow (laut <= 200 m)", or "Offshore deep (laut > 200m)", 'reservoir':"Sandstone" or "Carbonate", 'sourceRock':string, 'play':string,}]}	{ "status": "success" }
/showcase/list-well	GET	Get List of FieldName and WellName		{"status": "success"", data": array [{"fieldName": [], "wellName": [] }] }
/showcase/data	POST	Get Showcase Data	{ "fieldName": "string", "wellName": "string"	{"status":"success", "data": array [

			}	{"fieldName": string, "wellName": string, "wellArea": int, "wellLatitude": float, "wellLongitude": float, "wellOilVolume": float, "wellGasVolume": float, "wellWaterDepth": string, "wellReservoir": string, "wellSourceRock": string, "wellPlay": string }] }
/files	POST	Upload new showcase images	{ files: array [], foldername: string, pointer: string }	{ "status": "success" }
/files/lists/{pointer}	GET	Get list of folders and files of the	{ "pointer": string, "pathname": boolean	{ "status": "success", "data": array [

		showcase images based on pointer	}	{ "id": string, "name": string, "type": string, "children": array [] }] }
/file	GET	Get showcase images	{ "path": string }	Binary stream
Map Visualization (3D Viewer Map) - /api/v1/common				
/vtpviewer	POST	Upload VTP Files	{ files: string($binary), foldername: string, pointer: 'well-vtp', 'line-vtp', or 'surface-vtp' }	
/vtpviewer/info	POST	Add Info VTP Files	{ "path": string, "type": string, "center":[float,float], "geodata": [[float, float]] optional, "radius": float optional, "tilt": float optional, "long": float optional, "width": float optional, "length": float optional }	

/vtpviewer/getInfo	POST	Get Info VTP Files	{ “path”: “string” }	{ "status": "success", "result": { “filename":string, "type": “surface-vtp”, “well-vtp”, or “line-vtp”, "center": [float, float], "radius": float, “tilt”: float, “long”: float, “width”: float, “length”: float } }
/files/lists/	GET	Get list of folders and files of the VTP data based on pointers	{ “pointer”: array[string], “pathname”: boolean }	{ "status":"success", "data": array [{ "id": string, "name": string, "type": string, "children": array [] }]}

Table 5.3 List of Endpoint Error Codes

Endpoints	Methods	Error Codes	Response Details
/showcase	POST	409	Already exists
/vtpviewer	POST	415	Please make sure that it is a proper vtp file
/vtpviewer/getInfo	POST	404	File not found
/vtpviewer/info	POST	404	File not found
/files/lists/	GET	404	

5.4 User Interface

5.4.1 Map Visualization (3D Viewer Map) Page

The Map Visualization (3D Viewer Map) Page on first load when user choose '3D Viewer Map' in the vertical nav menu as shown Figure 5.3. The map shown is made using Leaflet. The tooltips on the map represents the location of the data. In this page, there is a Card on top of the map which is according to the design in Figure 4.7. Inside the card, there is Vuetify's text field, tree view, and button.

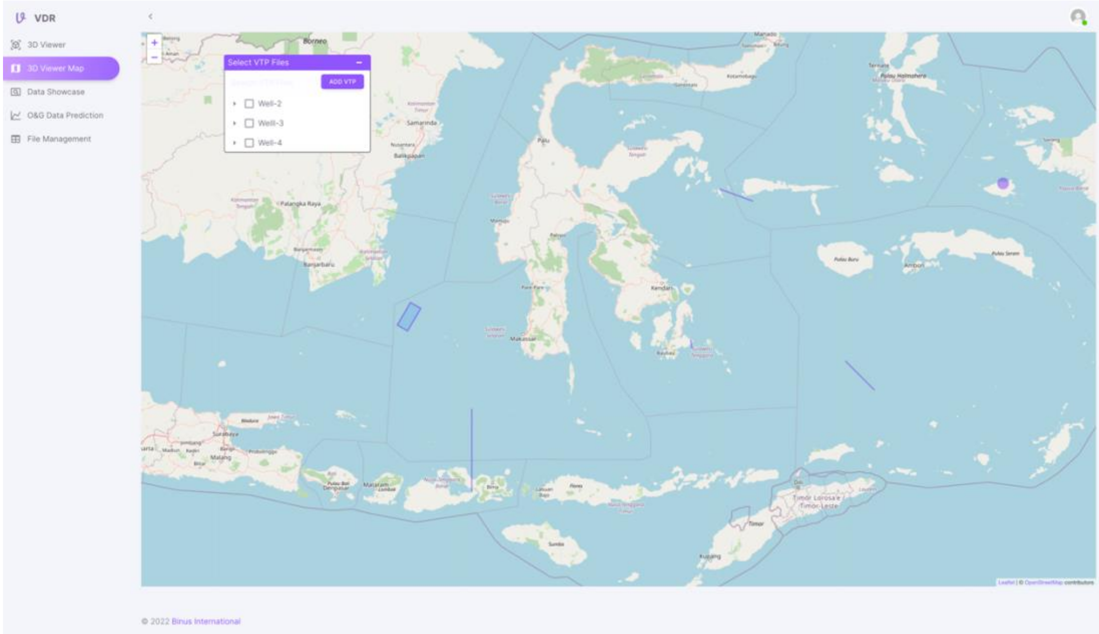


Figure 5.3 Map Visualization (3D Viewer Map) Page

The text field is used to search through the filter values. At the same time, the tree view allows the users to choose multiple values, including the data with parent and child structures. The filter values are the .vtp files of the wells. The tooltips on the map will darken their colors when the filter chosen as shown in Figure 5.4. The card is wrap using the *VueDraggableResizable* component to be moveable and resizable as shown in Figure 5.2.

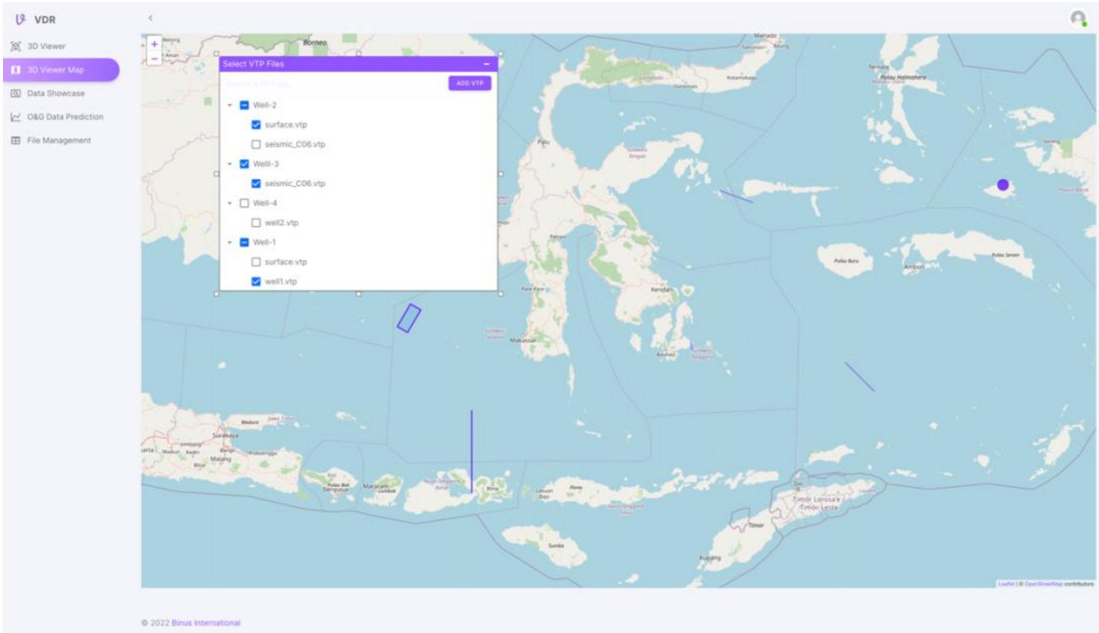


Figure 5.4 Map Visualization (3D Viewer Map) Page - On Filter

When a tooltip is click, two cards will show as depicted in Figure 5.3 and the tooltip color will change. One card will display the .vtp file chosen according to the tooltip clicked. Inside this card, an open-source library VTKjs is used to display the file. With VTKjs, the 3D object can be visualized interactively such as turning the object 360°, zoom in, and zoom out. The other card contains the widgets to change the style of the object such as change the color and color concentration of the object. In Figure 5.5, the tooltip chosen is circle which represent the well log of the wells in that location. The well log is in the form of 3D cylindrical object.

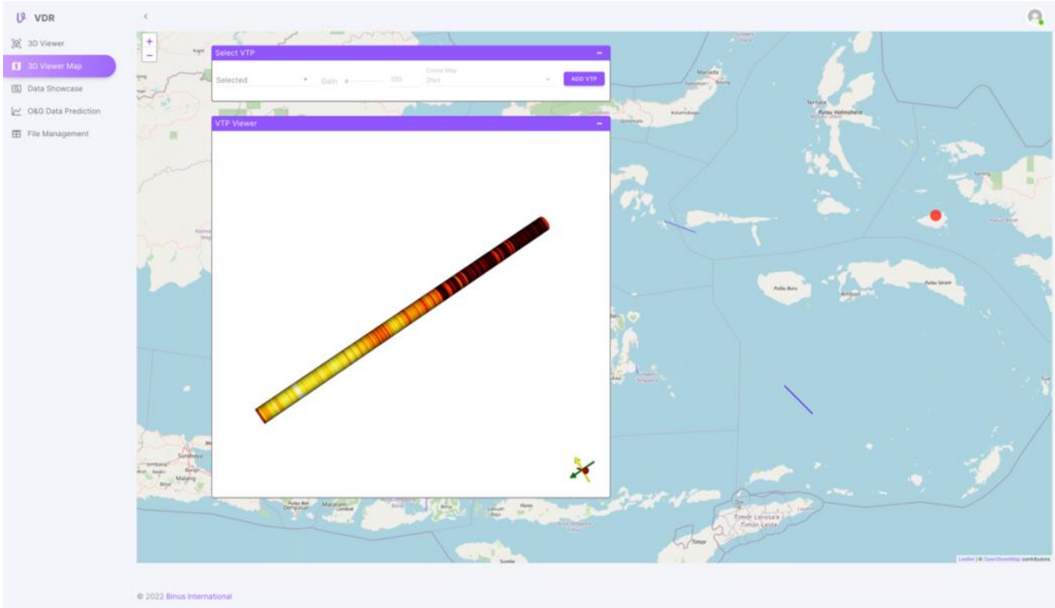


Figure 5.5 Map Visualization (3D Viewer Map) Page – 3D Well Log VTP shown in Card when Tooltip Clicked

When another tooltip is clicked, another 3D object will show in the card. In Figure 5.6, the tooltip clicked is in the form of line which represent the seismic line of the well in that location. The seismic line is in the form of 3D square object with line texture.

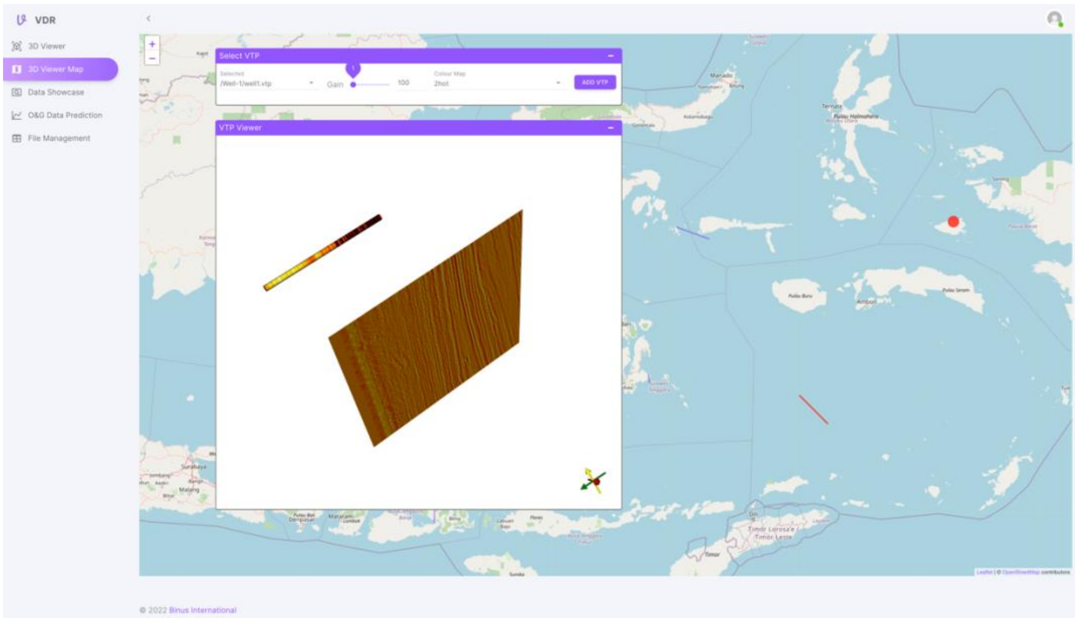


Figure 5.6 Map Visualization (3D Viewer Map) Page – 3D Seismic Line VTP added to the Card

In Figure 5.7, the tooltip clicked is in the form of rectangle which represent the subsurface of the well in that location. The surface is in the form of 3D square object with gradient texture.

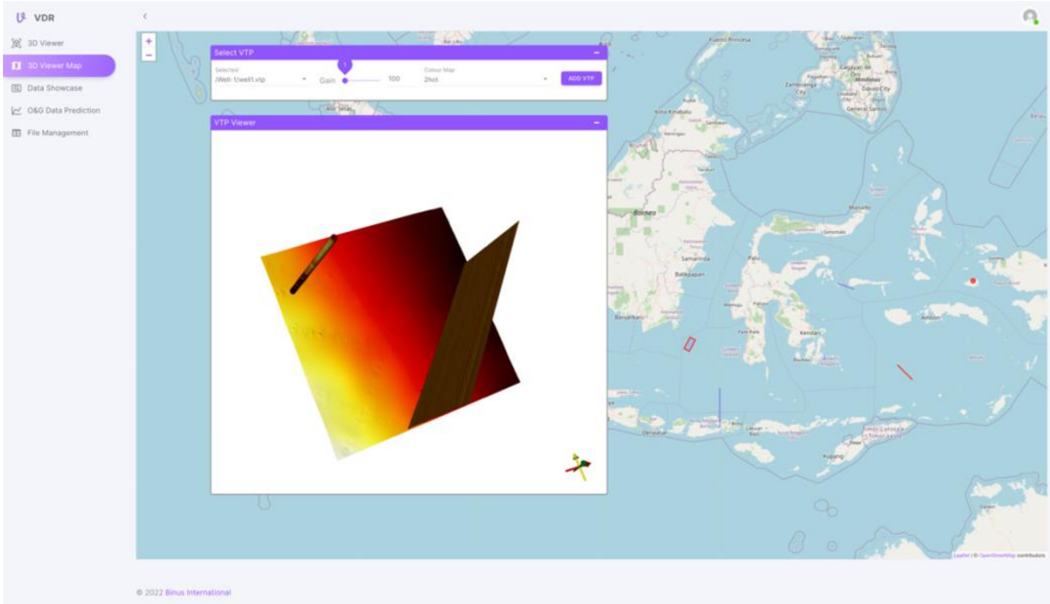


Figure 5.7 Map Visualization (3D Viewer Map) Page – 3D Surface VTP added to the Card

When color intensity ‘Gain’ is added up, the color of the object is darken as shown in Figure 5.8.

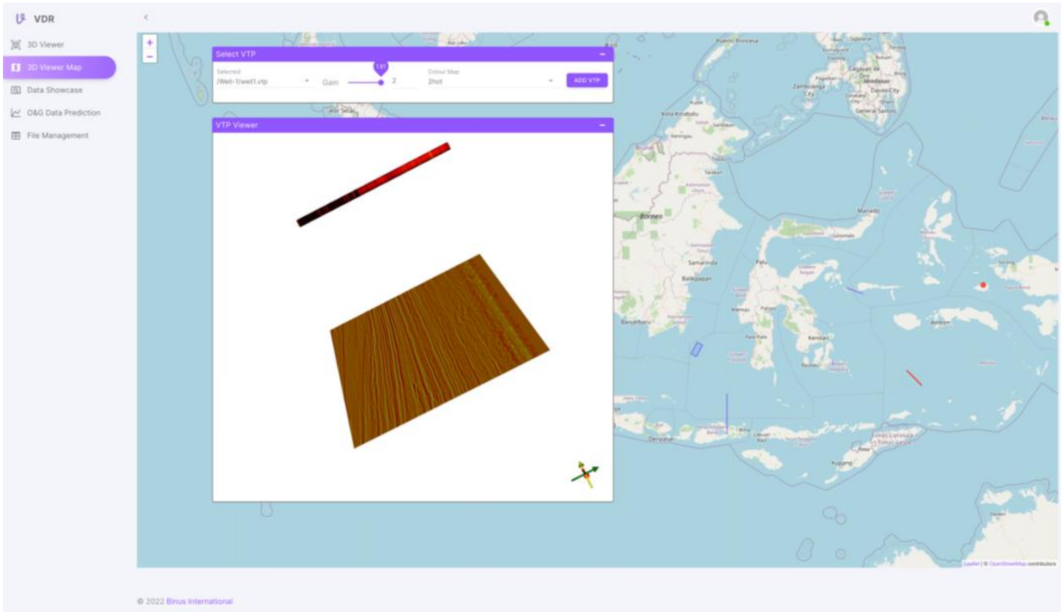


Figure 5.8 Gain of the 3D Well Log Increased

When color map is change, the color of the object change based on the chosen value as shown in Figure 5.9.

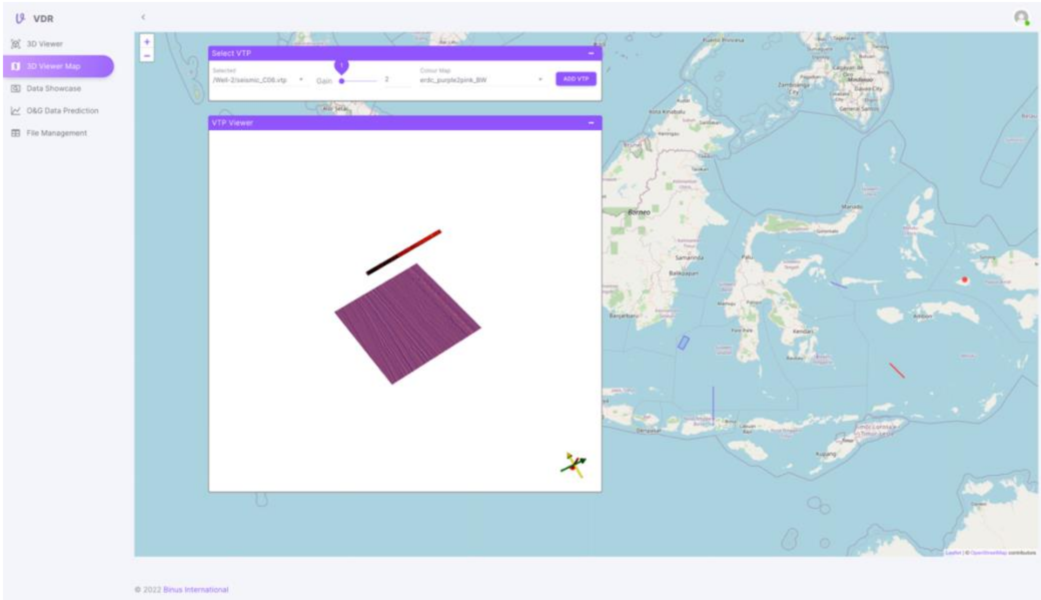


Figure 5.9 Map Visualization (3D Viewer Map) Page – Color Map of the 3D Seismic Line Changed

The two cards are also wrap using the *VueDraggableResizable* component so that it is moveable and draggable as illustrated in Figure 5.10. It is also minimizable as shown in Figure 5.11.

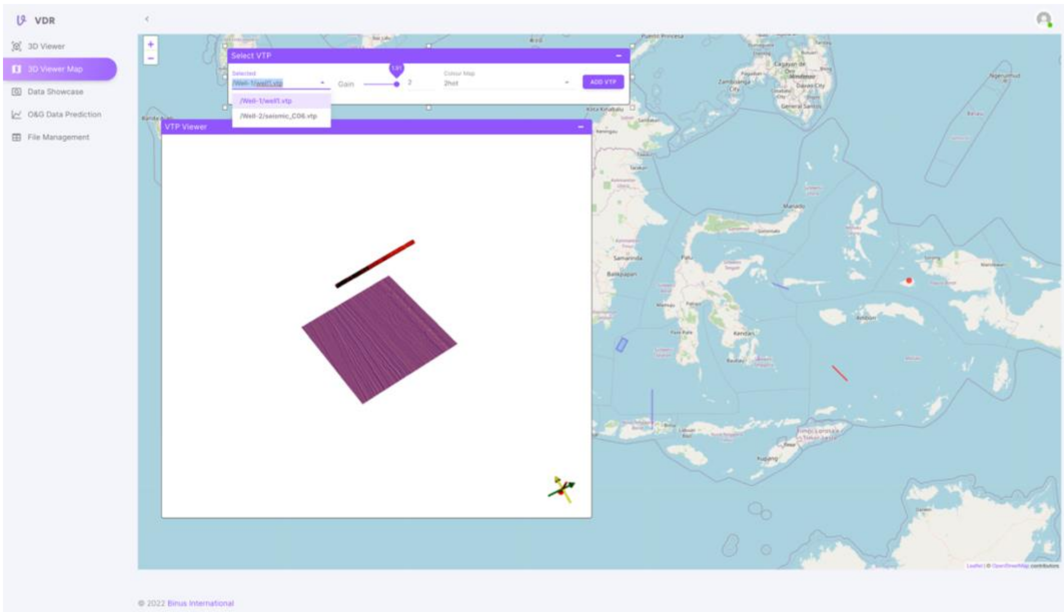


Figure 5.10 Map Visualization (3D Viewer Map) Page – Cards dragged and resized

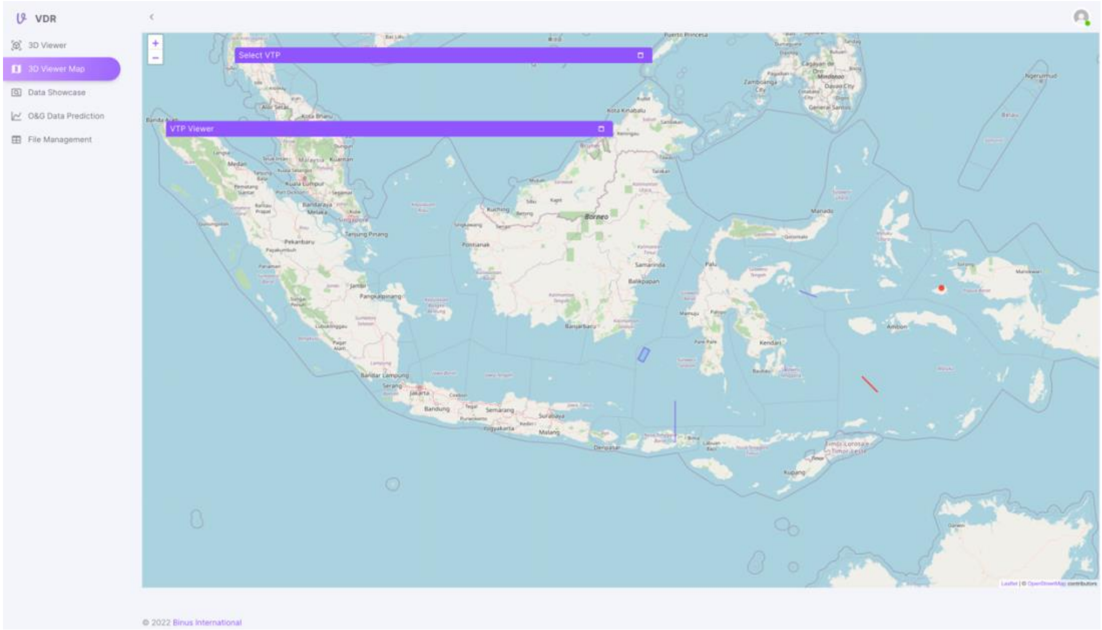


Figure 5.11 Map Visualization (3D Viewer Map) Page – Card Minimized

5.4.2 VTP Form Page

When the button ‘Add VTP’ in Figure 5.3 and 5.5 is click, user will be direct to VTP Form page. User will require to fill on the fields in this page as shown in Figure 5.12. User also need to choose the location of the well using the Leaflet map by double tapping a point on the map. User can also drag and drop the tooltip on the map to change the location of the well. There are certain fields that are hide out based on user selection of the ‘Select VTP type’ field.

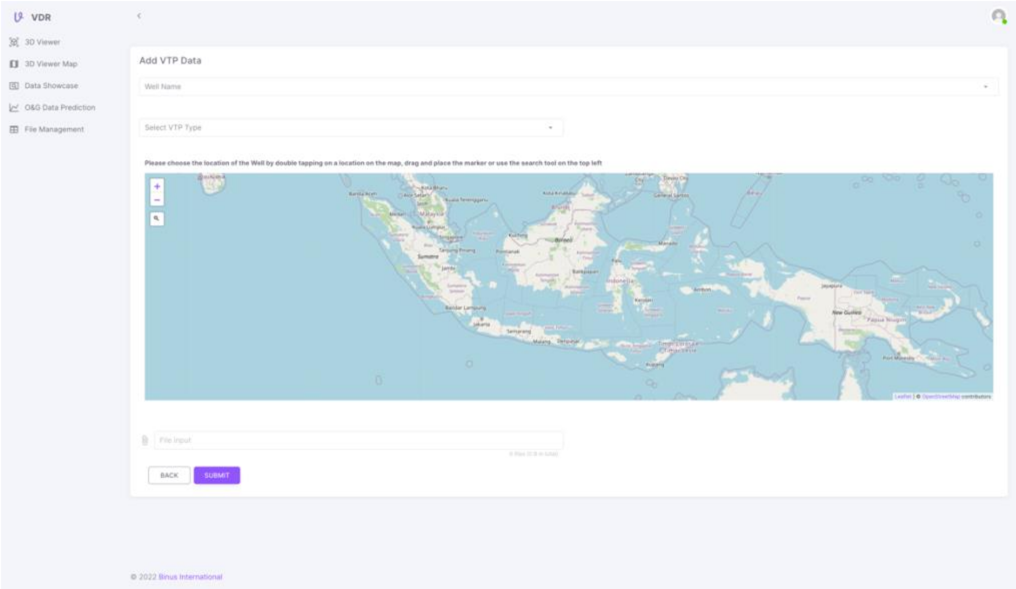


Figure 5.12 VTP Form Page

If user choose well-vtp, radius field will show where users are supposed to input the well radius as depicted in Figure 5.13.

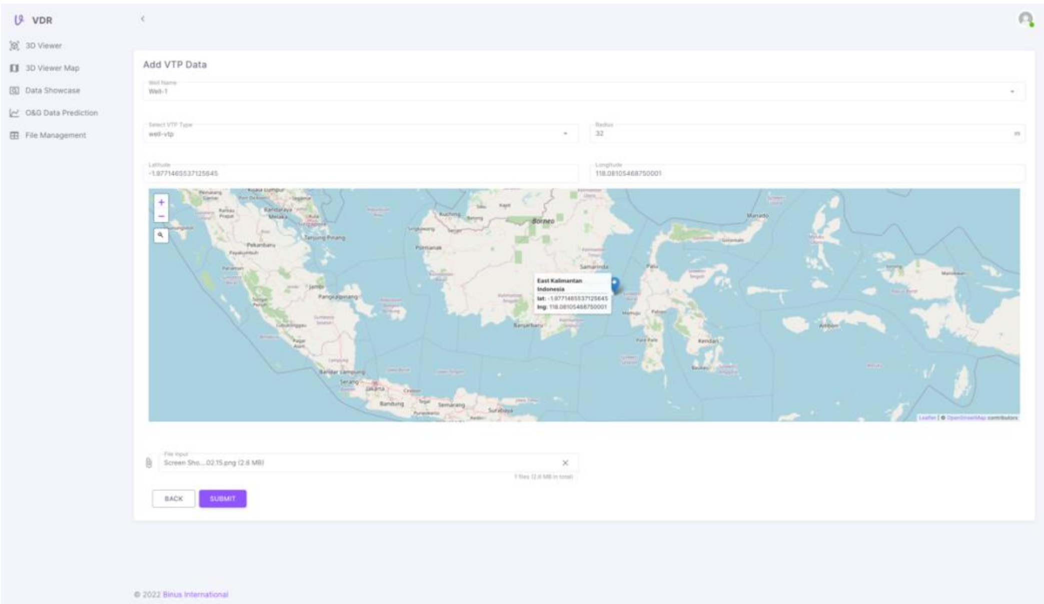


Figure 5.13 VTP Form Page - well-vtp chosen as VTP type

If user choose line-vtp, tilt angle and long fields will show as depicted in Figure 5.14. With the tilt angle and long, the author can measure the length of the seismic line on map and how tilted is the seismic line on map as shown in Figure 5.6. With latitude and longitude inputted by the user as the center of the seismic line. The equation to measure the latitudes and longitudes of the left end side and the right end side of the line is:

$$latR = latI + ((l/2)/110.574) * \cos((\pi / 180) * tA)), \quad 5.1$$

$$lngR = lngI + ((l/2)/110.574) * \sin((\pi / 180) * tA)), \quad 5.2$$

$$latL = latI - ((l/2)/110.574) * \cos((\pi / 180) * tA)), \quad 5.3$$

$$lngL = lngI - ((l/2)/110.574) * \sin((\pi / 180) * tA)). \quad 5.4$$

In the equations above, there common variables used which are *latI*, *lngI*, *l*, and *tA*. *latI* is the latitude inputted by the user and *lngI* is the longitude inputted by the user. Then *l* is the long (distance between right and left points) and *tA* is the tilt angle. The equation 5.1 is to find the right-side latitude, which is represent by symbol, *latR*. The equation 5.2 is to find the right-side longitude, which is represent by symbol, *lngR*. The equation 5.3 is to find the left side latitude, which is represent by symbol, *latL*. The equation 5.4 is to find the left side longitude, which is represent by symbol, *lngL*. The long, *l*, is divided by 2 as center to right/left is half the distance and divided again by 110.574 is to convert the km to degree.

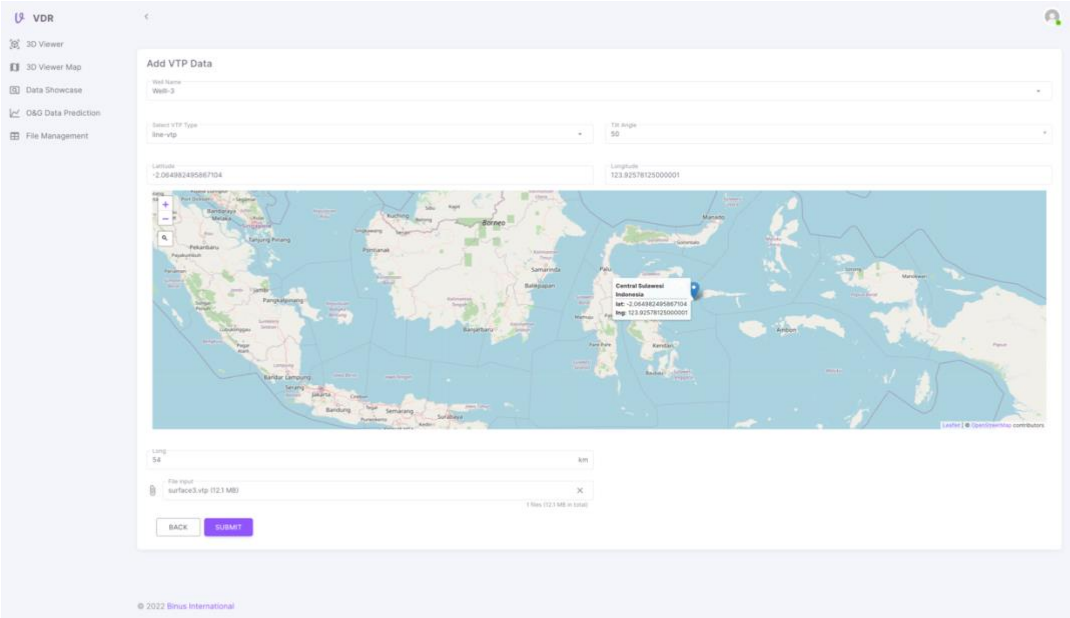


Figure 5.14 VTP Form Page - line-vtp chosen as VTP type

If user choose surface-vtp, tilt angle, length and width fields will show as depicted in Figure 5.15. With the tilt angle, length and width, the author can measure the latitudes and longitudes of the rectangle tooltip as the rectangle tooltip represent subsurface of the well. The rectangle tooltip is as shown in Figure 5.7. With latitude and longitude inputted by the user as the center of the subsurface. The equation to measure the latitudes and longitudes of the 4 corners of the rectangle is:

$$latN = latI + \left(\frac{\left(\frac{w}{2}\right)}{110.574}\right), \tag{5.5}$$

$$lngN = lngI - \frac{\frac{l}{2}}{111.32 * \cos\left(\left(\frac{\pi}{180}\right) * \left(\frac{\left(\frac{w}{2}\right)}{110.574}\right)\right)}, \tag{5.6}$$

$$latS = latI - \left(\frac{\left(\frac{w}{2}\right)}{110.574}\right), \tag{5.7}$$

$$lngS = lngI + \frac{\frac{l}{2}}{111.320 * \cos\left(\left(\frac{\pi}{180}\right) * \left(\frac{\left(\frac{w}{2}\right)}{110.574}\right)\right)}. \tag{5.8}$$

Similar to the previous set of equations, *latI* is the latitude inputted by the user and *lngI* is the longitude inputted by the user. The common variable in equation 5.5 – 5.8 is the *w* which are width inputted by the user. Aside from that, *latN* is the Northeast latitude in equation 5.5. *lngN* is the Northeast longitude in equation 5.6. latS is the Southwest latitude in equation 5.7. *lngS* is the Southwest longitude in equation 5.8.

After the equation applied, the latitudes and longitudes of the 4 corners of the rectangle will be:

- Corner 1: Southwest latitude, Northeast longitude
- Corner 2: Northeast latitude, Northeast longitude
- Corner 3: Northeast latitude, Southwest longitude
- Corner 4: Southwest latitude, Southwest longitude

Then there will be another equation to rotate the square based on the tilt angle inputted. The equation is applied to all the latitudes and longitudes to find the new latitudes and longitudes of the rectangle when rotated. The equation is:

$$\begin{aligned} latT = {} & \big(\cos(tA) * (latC - latI)\big) \\ & - (\sin(tA) * (lngC - lngI)) + latI, \end{aligned} \tag{5.9}$$

$$lngT = (\sin(tA) * \ (latC - latI)) \qquad 5.10$$
$$+ \left(\cos(tA) * (\ lngC - lngI)\right) + \ lngI.$$

The *latI*, *lngI*, and *tA* in equation 5.9 and 5.10 still represent the same things as previous equations. The *latI* is the latitude inputted by the user, *lngI* is the longitude inputted by the user, and the *tA* is the tilt angle. The common variables in equation 5.9 and 5.10 are the *latC* and *lngC*. The *latC* represent the latitude of each corner of the rectangle and the *lngC* represent the longitude of each corner of the rectangle.

The *latT* represent tilted corner latitude that want to be measure in equation 5.9 and the *lngT* represent tilted corner longitude that want to be measure in equation 5.10.

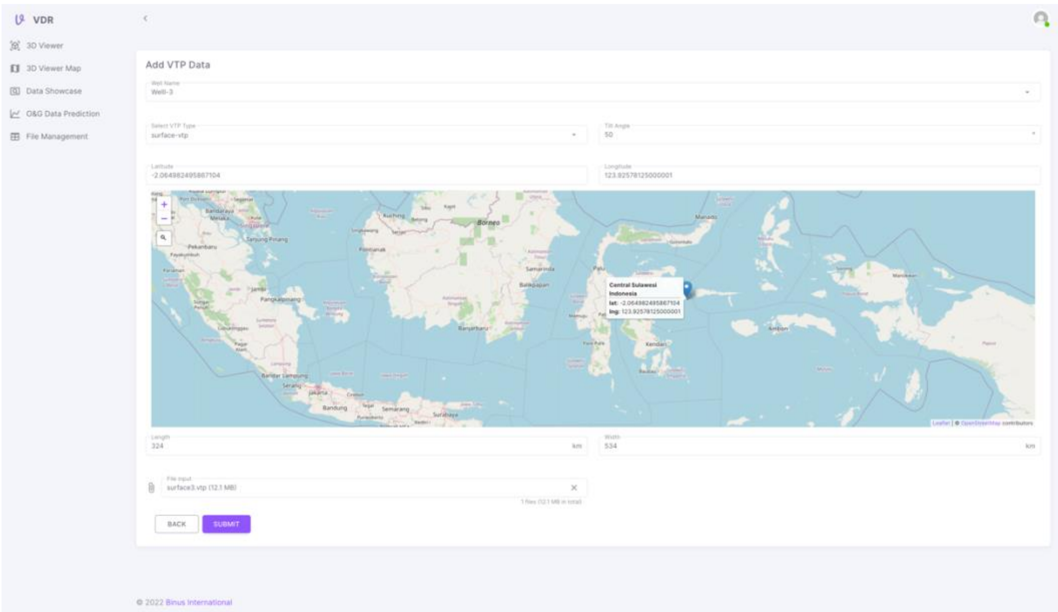


Figure 5.15 VTP Form Page - surface-vtp chosen as VTP type

After the user filled all the fields and pressed the Submit button, the user will need to wait while the application saves the data. There will be loading progress bar as shown in Figure 5.16. Aside from the submit button, the back button in this page will direct user to Map Visualization (3D Viewer Map) Page.

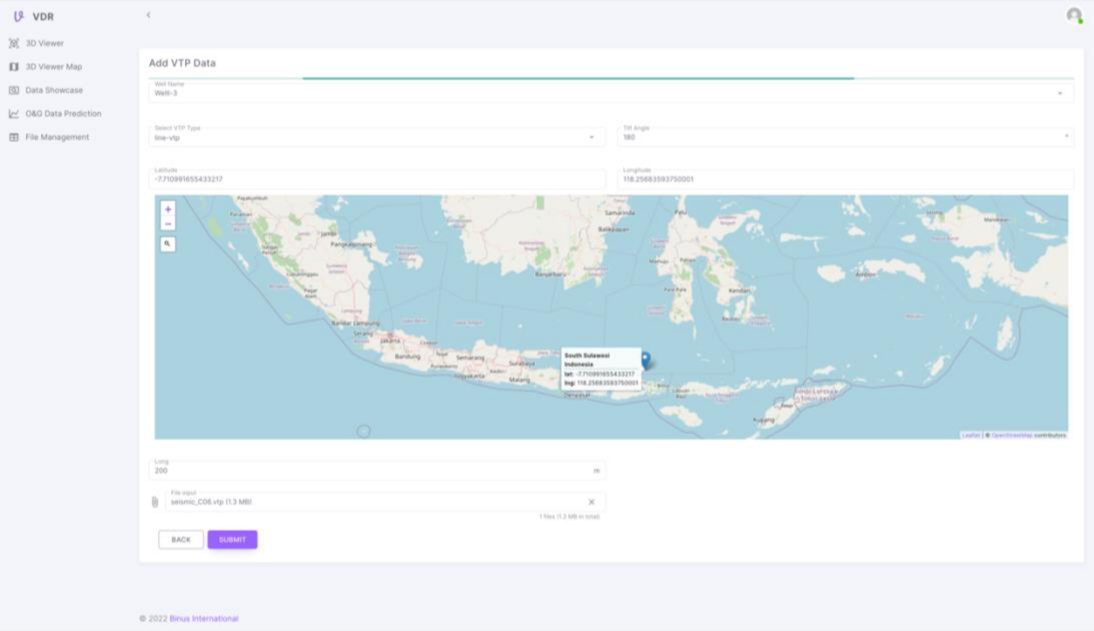


Figure 5.16 VTP Form Page - Loading to Save Data

After the application finish loading and successfully save data, there will be a confirmation card that confirm whether user want to add more VTP data or not as shown in Figure 5.17. When user tap ‘Yes’, the VTP Form Page will reload. When user tap ‘No’, user will go back to Map Visualization (3D Viewer Map) Page.

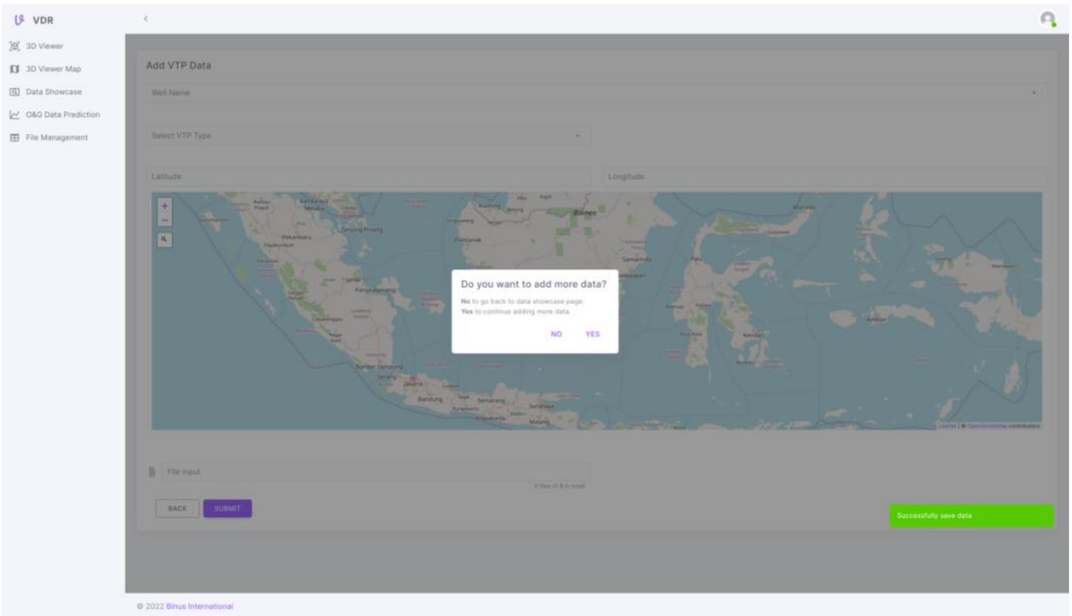


Figure 5.17 VTP Form Page - Confirmation Card

5.4.3 Map Data Showcase Page

The Map Data Showcase Page on first load when user choose ‘Data Showcase’ in the vertical nav menu as shown Figure 5.18. The map shown is also made using Leaflet. The tooltips on the map represents the location of the well. In this page, there is two Cards and a button on top of the map which is according to the design in Figure 4.17.

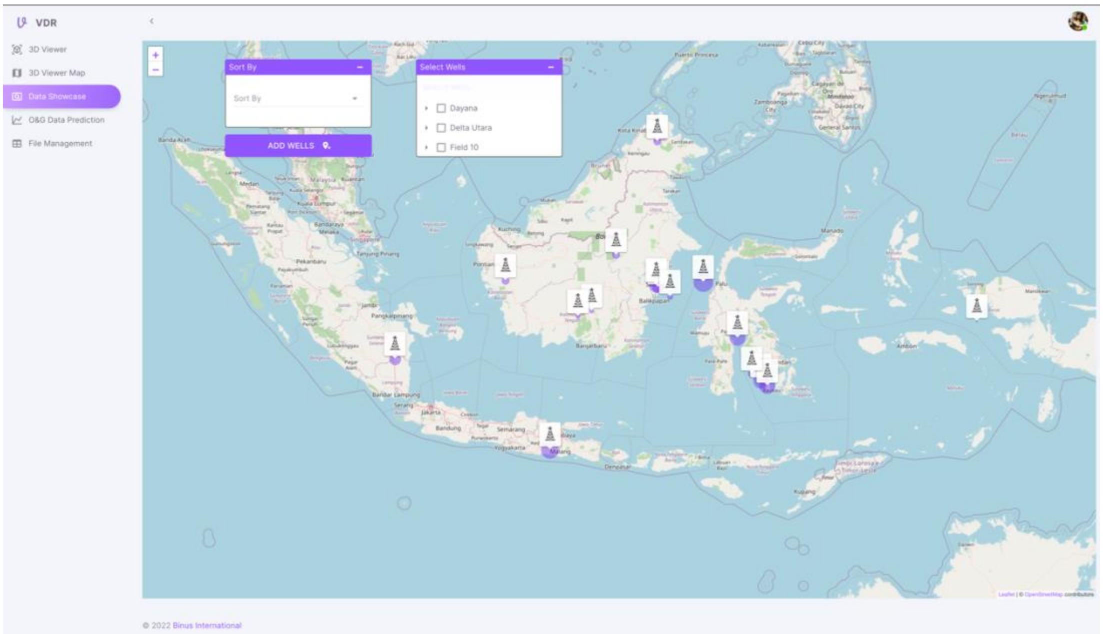


Figure 5.18 Map Data Showcase Page

In one of the cards, there is Vuetify’s text field and tree view. The text field is used to search through the filter values. At the same time, the tree view allows the users to choose multiple values, including the data with parent and child structures. The filter values are the fields and wells of oil and gas drill locations. The tooltips on the map will change their colors when the filter chosen as shown in Figure 5.19. This card is also wrap with the *VueDraggableResizable* component as shown in Figure 5.19.

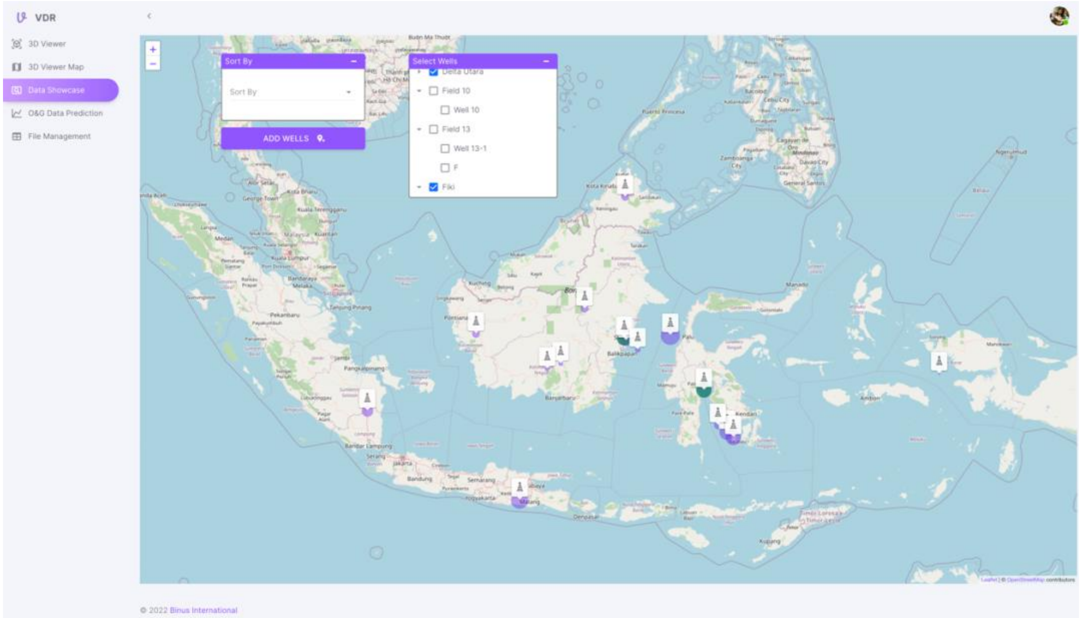


Figure 5.19 Map Data Showcase Page - On Filter

In the other card, there is Vuetify's autocomplete component. The autocomplete component provides a dropdown with ability to search the value of the dropdown and other various abilities. One of the other abilities of autocomplete components used for this feature is clearable, which allows users to clear the value of the sort when they do not want to implement it. The values that can be chosen are *Oil Volume* and *Gas Volume*, where the tooltips on the map will be sorted based on the volume chosen. The color surrounding the circle tooltips will have darker shades of red when the volume is high and light shades of red when the volume is low. The range of the shades can be seen in Figure 5.20. Similar with the other cards, this card is also wrap with the *VueDraggableResizable* component.

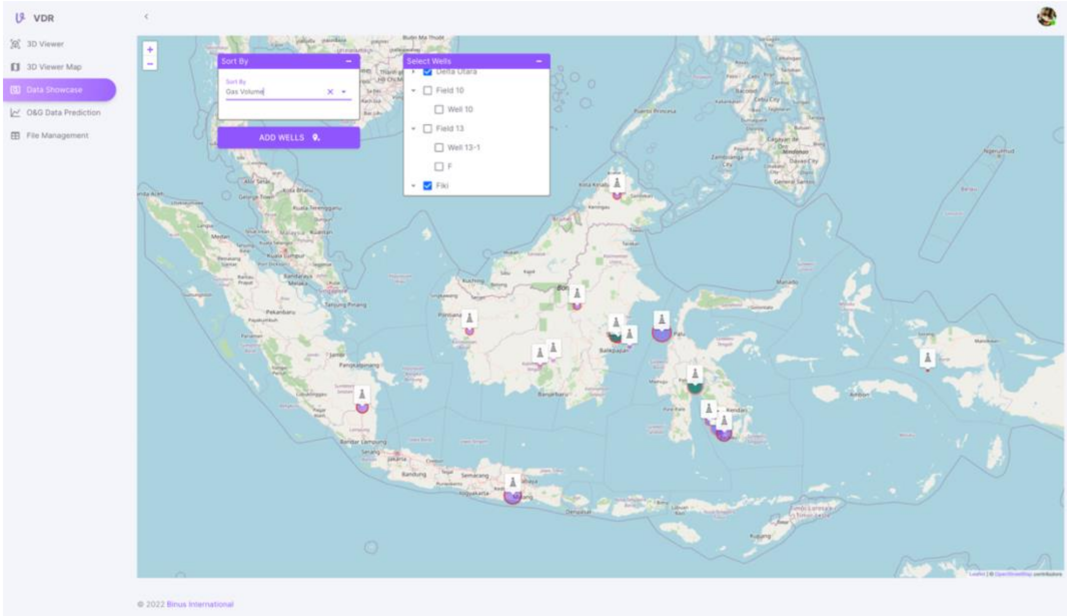


Figure 5.20 Map Data Showcase Page - On Filter and Sort

As the tooltips represents the location of the wells. When a tooltip is clicked, a card overlay will show up. The card overlay contains checkboxes where users can choose which showcase images they want to see, as shown in Figure 5.21. Aside from the checkboxes, a close button and a proceed button is also present. The close button will close the overlay. While the proceed button allow user to see overlay with the showcase images and showcase data. The proceed button is initially disabled to be click as there is no checkboxes selected yet.

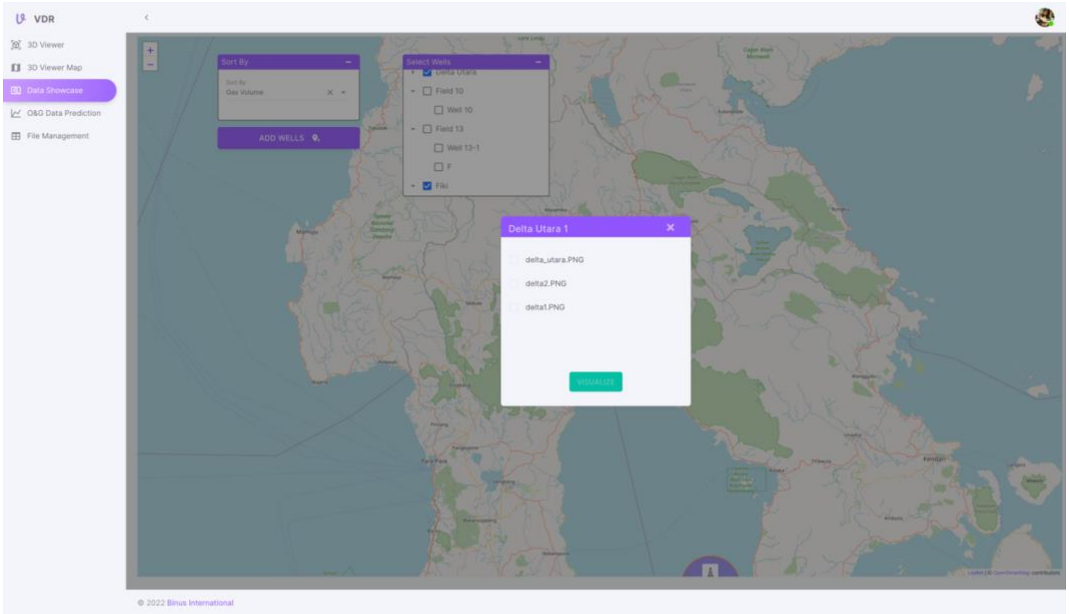


Figure 5.21 Map Data Showcase Page - Overlay Shown after a Tooltip Clicked

After user selected one or more checkboxes, the proceed button become clickable as shown in Figure 5.22.

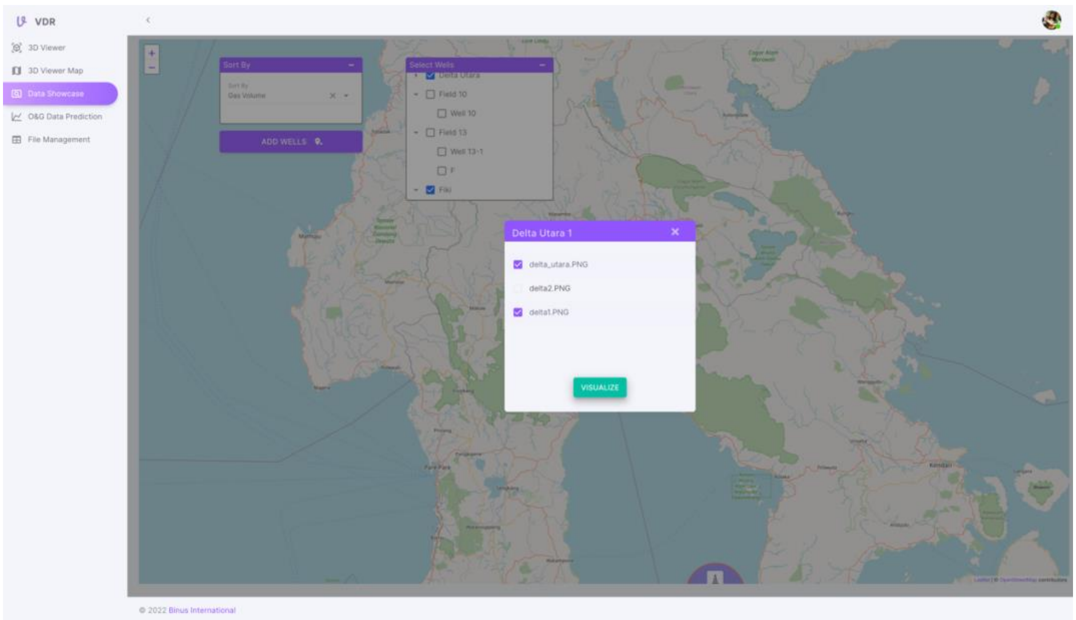


Figure 5.22 Map Data Showcase Page - Proceed Button Clickable

After user tap the proceed button, another overlay with the showcase images and showcase data will show up as shown in Figure 5.23. In this overlay, there are also download button and close button. When user click the close button, the overlay will close.

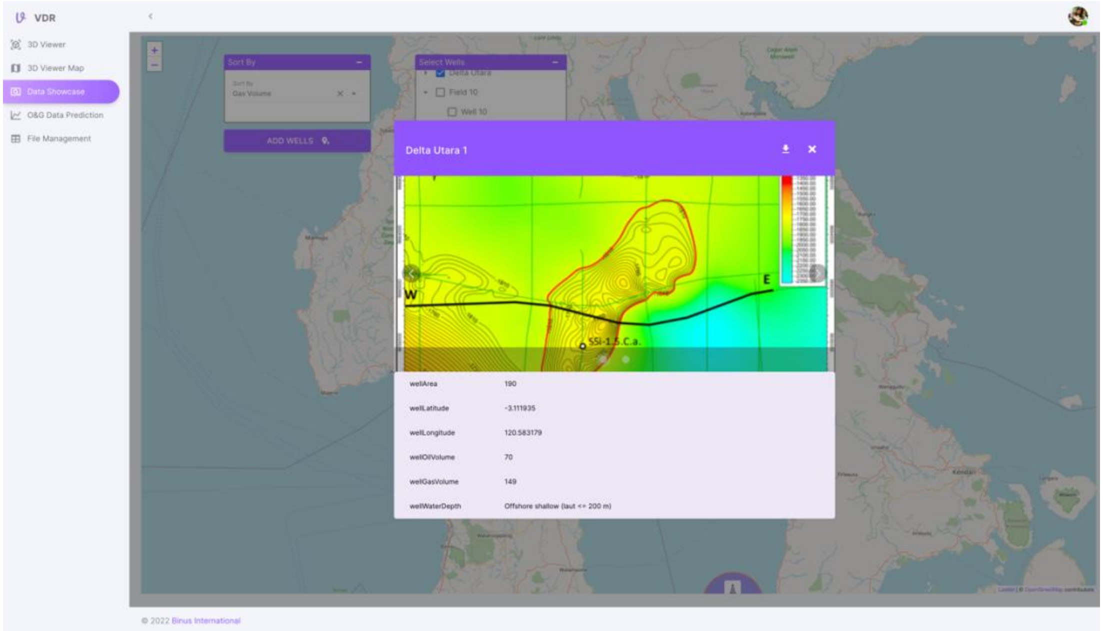


Figure 5.23 Map Data Showcase Page - Showcase Overlay

After user click the download button, the application will generate a pdf file with the showcase images and data. The content of the pdf file is as shown in Figure 5.25.

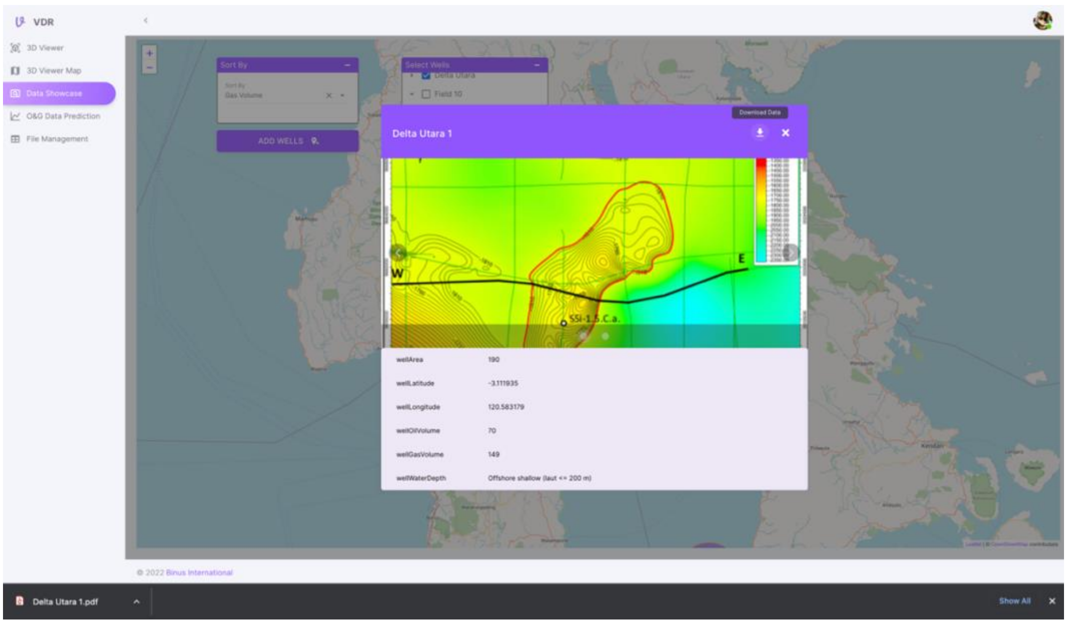


Figure 5.24 Map Data Showcase Page - Download Showcase Data

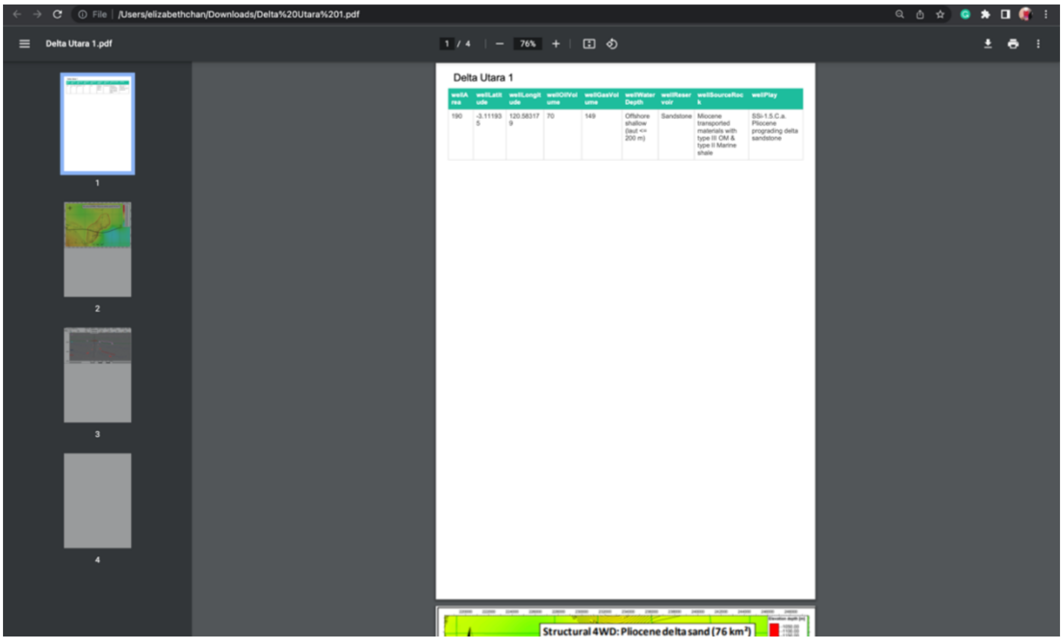


Figure 5.25 Map Data Showcase Page - Example of PDF File Downloaded

5.4.4 Data Showcase Form Page

When the ‘Add Wells’ button in Map Data Showcase Page is clicked, user is redirect to Data Showcase Form Page as shown in Figure 5.26.

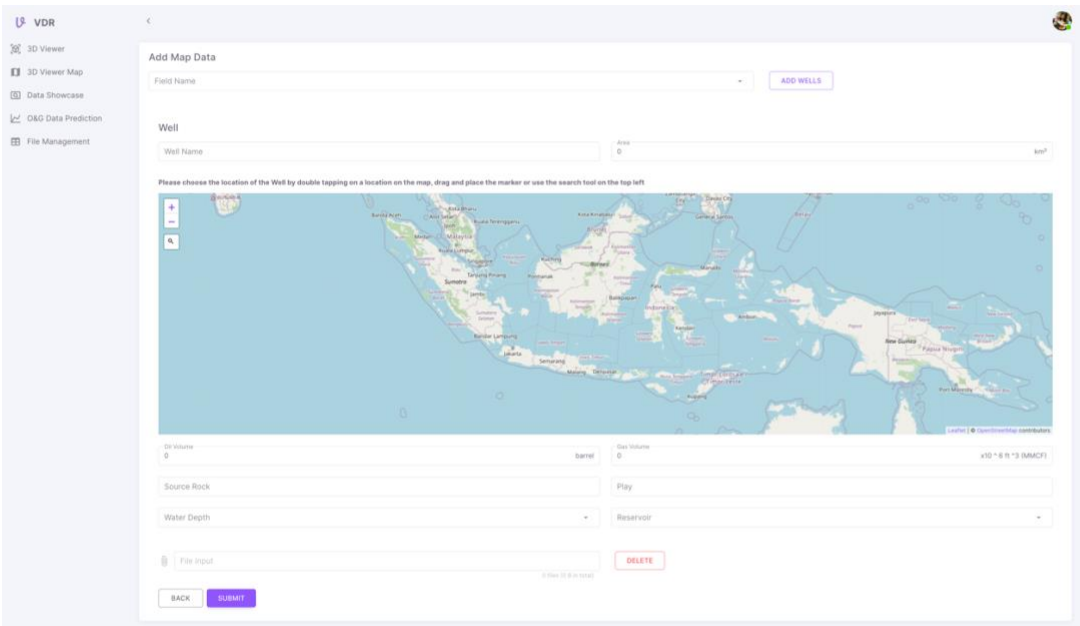


Figure 5.26 Data Showcase Form Page

When all the fields in Map Data Showcase Page are filled, the page will look like Figure 5.27.

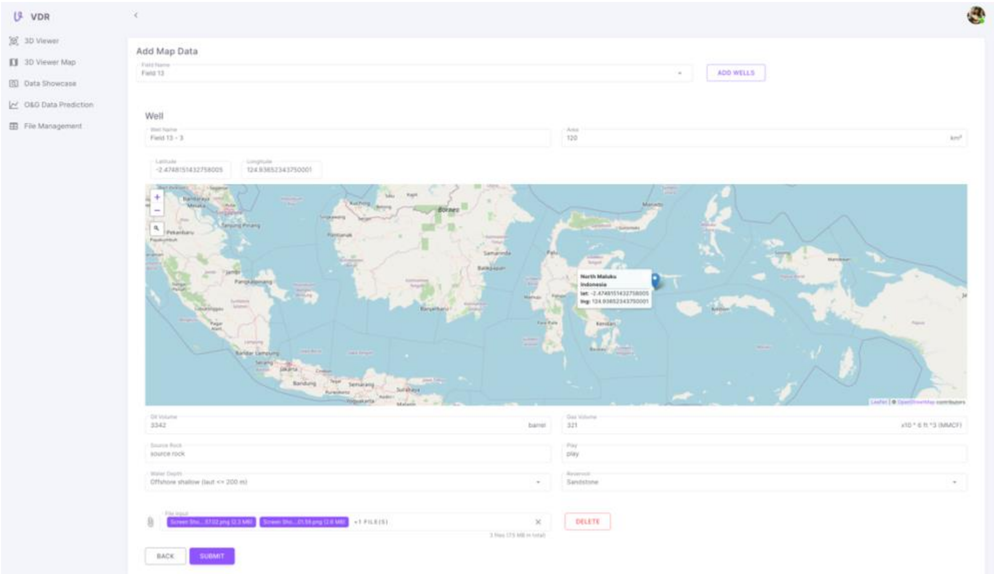


Figure 5.27 Data Showcase Form Page - All Fields Filled

When 'Add Wells' button in Data Showcase Form Page is clicked, a new set of Well fields will show as depicted in Figure 5.28. When 'Delete' button is clicked, the set of Well fields will be deleted.

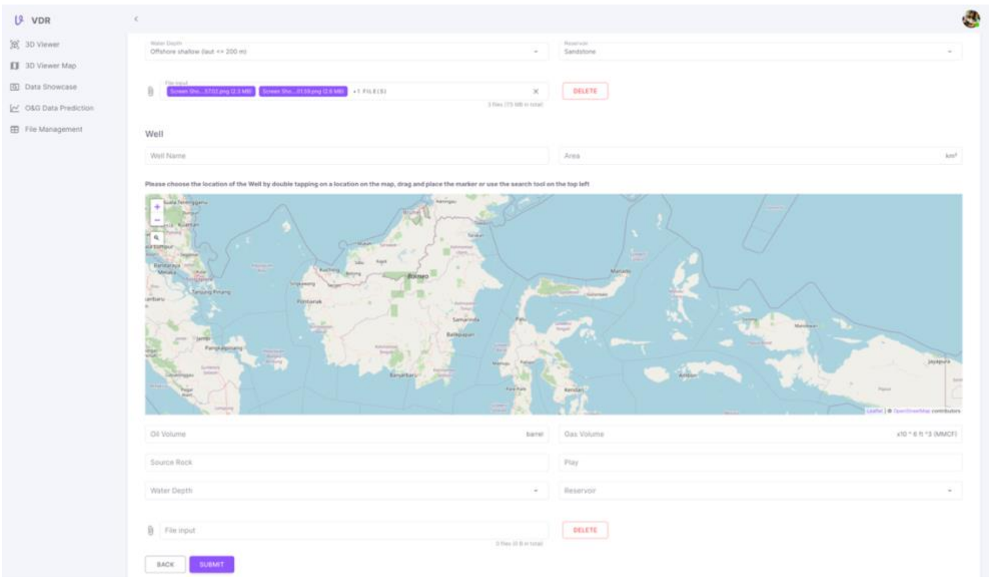


Figure 5.28 Data Showcase Form Page - A Set of Well Fields Added to the Page

Similar to the map on the VTP Data Form Page, user need to choose the location of the well using the Leaflet map by double tapping a point on the map. User can also drag and drop the tooltip on the map to change the location of the well. Additionally, user can use the search tool to find the location of the well as shown in Figure 5.29.

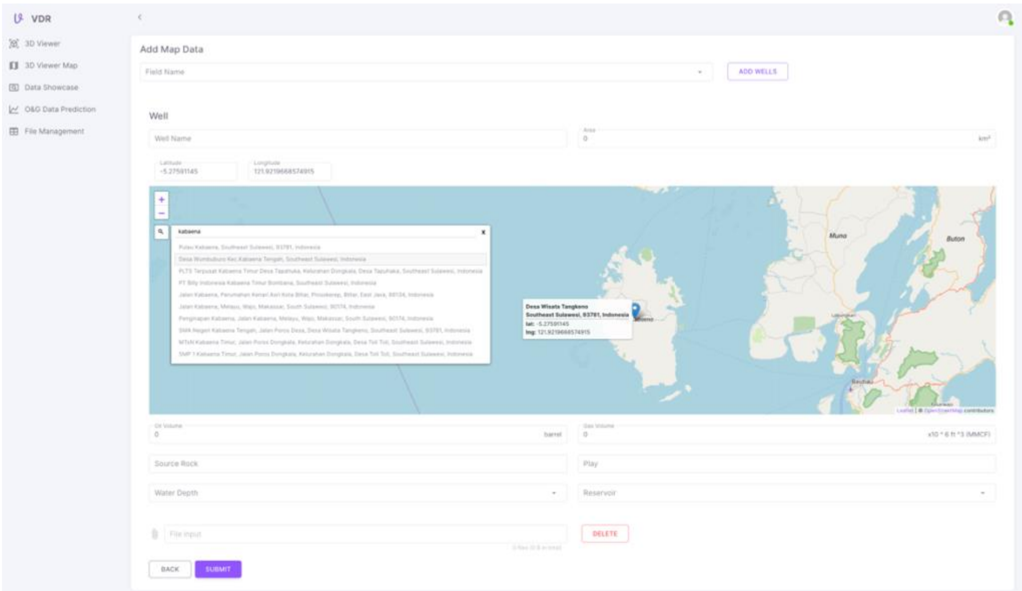


Figure 5.29 Data Showcase Form Page - Search Tool on the Map

Same as the VTP Form page, there will be a confirmation card that confirm whether user want to add more VTP data or not after user tap Submit button and the application successfully save data, as shown in Figure 5.30. When user tap 'Yes', the VTP Form Page will reload. When user tap 'No', user will go back to Map Data Showcase Page.

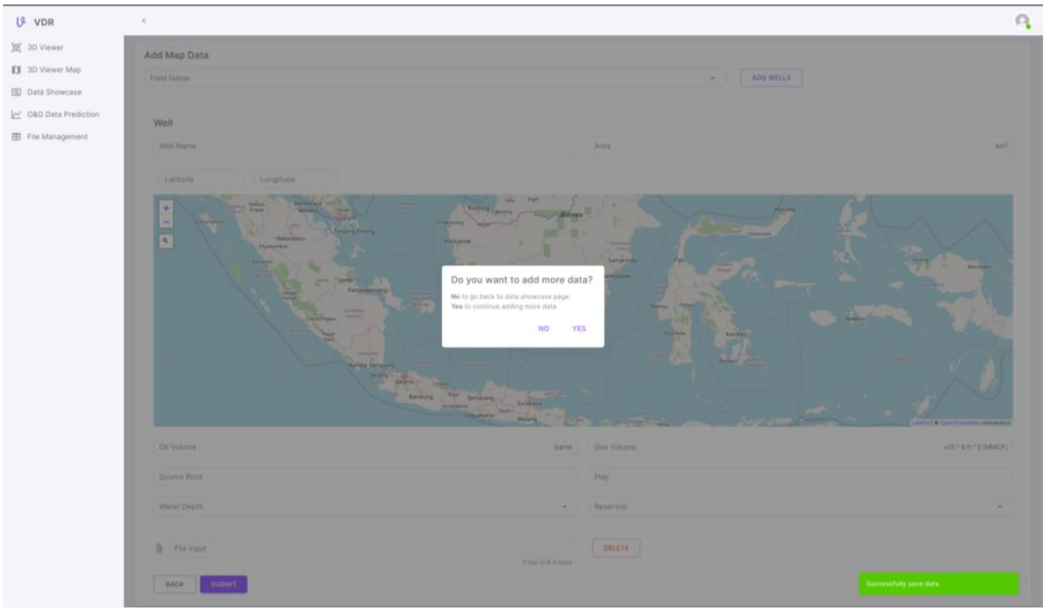


Figure 5.30 Data Showcase Form Page - Confirmation Page

CHAPTER 6
DISCUSSION

6.1 Development

The development period of the VDR website application lasted four months. The development was done in sprints as mentioned Section 5.1. Throughout the sprints, the design and implementation of the features that the author is responsible for already underwent several changes. Before changes are made, the author, author's team and product owner will have discussions first to ensure that the changes are possible. The changes are done in order to meet up to the standard of the application for the client and users. After every change, the author and team will present it to the product owner and get the product owner's approval. Fortunately, the cost of the development is trivial as the author use OSS.

6.2 Evaluation

The author and team have fulfilled every request suggested by the product owner and meet the satisfaction of the product owner. Unfortunately, the author and team don't have enough time to conduct the UI testing due to the limited duration time. However, the author and team had demonstrated the application with all features integrated to the client and product owner. The client and product owner are quite satisfied with the progress and UI. The demonstration was conducted through a zoom meeting.

As the development still on initial phase, it is believed that there will be major enhancement in future phase to develop the application entirely so that the application can be useful for oil and gas industries in Indonesia.

6.3 Constraints

The constraints that the author encounter in developing the VDR website application until the end of the development are:

- Responsiveness

  There are some components of the application that already responsive. However, the author did not manage to make some major components to be responsive. Therefore, the website can only be open using big screen device like laptop so that the components will not disorganized.

- Detail existing applications reference

  As most of the existing applications are not free of charge, the author can only reference from demo provided by the applications which are very limited. Aside from that, the author can only envisage the application based on the information given by the client. Therefore, the author believed that an access to existing application and thorough observation is needed to ensure the design of the VDR website application is suitable for the users.

- Simplicity

  According to the client, there are too many fields in the Data Showcase Form page which may inconvenience the user. However, the author has no option as the client cannot share classified template on how the data is normally store.

CHAPTER 7

CONCLUSION AND RECOMMENDATION

This chapter conclude the achievement achieved at the end of developing the application and provide some recommendations that can applied in future works.

7.1 Conclusion

After the development of the application and demonstration to client are done, the author concluded that the client is satisfied with the progress of the website application. The client consent that the application has been delivered according to the prototype requirements and scope assigned to the author and team. Additionally, the author and team has managed to reach the aims of developing this application which are:

- Build a customizable VDR website application,
- visualizing the oil and gas production data including the well log, seismic line, and well subsurface data with the implementation of GIS,
- showcasing the oil and gas information with the integration of GIS,
- come up with a predictive model that can predict the oil and gas production, and
- visualizing the oil and gas prediction data.

As the author and team manage to achieve the aims, the author and team have also achieved to cater the benefit of the VDR website application.

7.2 Recommendations

Based on the result of the development and discussion with product owner and client, there are some things that are recommended to be improve in the futures:

- In the Data Showcase Form page, it will be better for users to input data with excel and csv file rather than inputting all the fields one by one. However, this is possible if the client provides classified template of storing the oil and gas data.
- In the Map Visualization (3D Viewer Map) Page and Map Data Showcase Page, some of the components are not responsive enough to be open in mobile devices or small screen devices. Therefore, in the future, it is preferred that the components can be responsive so that the application can increase the number of device compatibilities.

REFERENCES

[1] U. Ali, "The history of the oil and gas industry from 347 AD to today," OFFSHORE TECHNOLOGY, 7 March 2019. [Online]. Available: https://www.offshore-technology.com/comment/history-oil-gas/#:~:text=The%20modern%20history%20of%20the,thicker%20oil%20suitable%20for%20lubrication.

[2] "Why oil is important," UK Oil & Gas PLC, [Online]. Available: https://www.ukogplc.com/page.php?pID=74#:~:text=Oil%3A%20lifeblood%20of%20the%20industrialised,people%20all%20over%20the%20world.

[3] D. Yergin, The New Map: Energy, Climate, and the Clash of Nations, New York: Penguin Group and U.S. Council on Economic Advisors, 2020.

[4] P. Yasmin, "6 Pemanfaatan Minyak Bumi bagi Kehidupan, Apa Saja?," detikFinance, 09 February 2021. [Online]. Available: https://finance.detik.com/energi/d-5367484/6-pemanfaatan-minyak-bumi-bagi-kehidupan-apa-saja.

[5] PwC Indonesia, "Oil and Gas in Indonesia," September 2019. [Online]. Available: https://www.pwc.com/id/en/energy-utilities-mining/assets/oil-and-gas/oil-gas-guide-2019.pdf.

[6] BADAN PUSAT STATISTIK, "STATISTIK PERTAMBANGAN MINYAK DAN GAS BUMI 2014 - 2019," p. 12, 2020.

[7] Y. Fan, K. Gao, J. Chen, W. Li and Y. Zhang, "Low-cost PMMA-based microfluidics for the visualization of enhanced oil recovery," *Oil & Gas Science and Technology – Revue d'IFP Energies nouvelles,* vol. 73, p. 26, 2018.

[8] INT, "Building an Upstream Data Management and Visualization Application," INT, [Online]. Available: https://www.int.com/customer_stories/pertamina/.

[9] Geodwipa Teknika Nusantara, "Geodwipa Teknika Nusantara," [Online]. Available: https://ptgtn.com/. [Accessed 5 April 2022].

[10] A. A. Purwita, Interviewee, *Oil and Gas Companies.* [Interview]. 8 March 2022.

[11] E. Ali, "Geographic Information System (GIS): Definition, Development, Applications & Components," 2020.

[12] B. Harrison, "The data room," *Developments in Petroleum Science,* vol. 69, pp. 21-26, 2020.

[13] M. Lewis, "The Importance of a Virtual Data Room Index and Folder Structure," DealRoom, 2 March 2022. [Online]. Available: https://dealroom.net/blog/the-importance-of-a-virtual-data-room-index-and-folder-structure. [Accessed 26 March 2022].

[14] Lynx Information Systems, "Lynx Virtual Data Rooms," Lynx Information Systems, [Online]. Available: http://www.lynxinfo.co.uk/vdr.html. [Accessed 26 March 2022].

[15] "Lynx iSeisview and iLogview Demo," Lynx Information Systems, [Online]. Available: https://data.lynxinfo.co.uk/apps/iseisview/nethpublic#zoom=9&lat=7196871.98607&lon=449014.9111&layers=0B000TTTFTT. [Accessed 26 March 2022].

[16] SIZLE RESOURCES TEAM , "What Is A Virtual Data Room?," SIZLE, June 2022. [Online]. Available: https://sizle.io/what-is-a-virtual-data-room/. [Accessed 30 June 2022].

[17] "RATS - FAQ," The Iowa Institute for Biomedical Imaging, [Online]. Available: https://www.iibi.uiowa.edu/rats-faq#:~:text=VTP%20is%20a%20file%20format,surface%20models%20into%20binary%20images. [Accessed 30 June 2022].

[18] Boston University Information Services & Technology, "Using VTK to Visualize Scientific Data (online tutorial)," [Online]. Available: https://www.bu.edu/tech/support/research/training-consulting/online-tutorials/vtk/. [Accessed 30 June 2022].

[19] G. Farkas, "Applicability of open-source web mapping libraries for building massive Web GIS clients," *Journal of Geographical Systems,* vol. 19, no. 3, pp. 273-295, 2017.

[20] A. Skopeliti and L. Stamou, "Online map services: Contemporary cartography or a new cartographic culture?," *ISPRS International Journal of Geo-Information,* vol. 8, no. 5, 2019.

[21] Vue Leaflet, "Introduction," [Online]. Available: https://vue2-leaflet.netlify.app/components/. [Accessed 2 April 2022].

[22] fega, "vue2-leaflet-geosearch," [Online]. Available: https://github.com/fega/vue2-leaflet-geosearch. [Accessed 9 June 2022].

[23] C. Maxwell, "The Evolution of Website Development," terrostar, 5 November 2020. [Online]. Available: https://terrostar.com/the-evolution-of-website-development/.

[24] " History of the Web," WORLD WIDE WEB FOUNDATION, [Online]. Available: https://webfoundation.org/about/vision/history-of-the-web/.

[25] M. Storm, "6 Types of Web Design to Consider for Your Website," WebFX, 14 June 2021. [Online]. Available: https://www.webfx.com/blog/web-design/types-of-web-design/.

[26] F. Almeida and J. Monteiro, "The role of responsive design in web development," vol. 14, no. 2, pp. 48-65, 2017.

[27] M. Lozhko, "Web Application Architecture: Best Practices and Guides," LANARS, [Online]. Available: https://lanars.com/blog/web-application-architecture-101. [Accessed 28 March 2022].

[28] altexsoft, 25 July 2019. [Online]. Available: https://www.altexsoft.com/blog/engineering/web-application-architecture-how-the-web-works/. [Accessed 28 March 2022].

[29] "What is Frontend Development?," Techslang, [Online]. Available: https://www.techslang.com/definition/what-is-frontend-development/.

[30] Y. K. Xing, J. P. Huang and Y. Y. Lai, "Research and Analysis of the Front-end Frameworks and Libraries in E-Business Development," *ACM International Conference Proceeding Series,* pp. 68-72, 2019.

[31] Vue.js, "Comparison with Other Frameworks," [Online]. Available: https://v2.vuejs.org/v2/guide/comparison.html?redirect=true. [Accessed 1 April 2022].

[32] "Overview," Vue.js, [Online]. Available: https://v1.vuejs.org/guide/overview.html. [Accessed 13 March 2022].

[33] "Materio – Free Vuetify Vuejs Admin Template," ThemeSelection, [Online]. Available: https://themeselection.com/products/materio-free-vuetify-vuejs-admin-template/. [Accessed 13 March 2022].

[34] MATERIAL DESIGN, [Online]. Available: https://material.io/. [Accessed 13 March 2022].

[35] J. Leider, "Why Vuetify?," Vuetify, [Online]. Available: https://vuetifyjs.com/en/introduction/why-vuetify/. [Accessed 13 March 2022].

[36] A. Andrews, "Material Design Icons," [Online]. Available: https://materialdesignicons.com/. [Accessed 13 March 2022].

[37] M. Meng, S. Steinhardt and A. Schubert, "Application Programming Interface Documentation: What Do Software Developers Want?," *Journal of Technical Writing and Communication,* vol. 48, p. 295–330, 2018.

[38] IBM Cloud Education, "Application Programming Interface (API)," IBM, 19 August 2020. [Online]. Available: https://www.ibm.com/cloud/learn/api. [Accessed 30 June 2022].

[39] IBM Cloud Education, "REST APIs," IBM, 6 April 2021. [Online]. Available: https://www.ibm.com/cloud/learn/rest-apis. [Accessed 30 June 2022].

[40] "Introducing JSON," [Online]. Available: https://www.json.org/json-en.html. [Accessed 30 June 2022].

[41] "What is open source?," Opensource.com, [Online]. Available: https://opensource.com/resources/what-open-source. [Accessed 12 March 2022].

[42] M. Michels, "A Brief History of Open Source," 31 May 2021. [Online]. Available: https://maximilianmichels.com/2021/history-of-open-source/. [Accessed 11 March 2022].

[43] M. Asay, "Microsoft may be the world's largest open source contributor, but developers don't care–yet," TechRepublic, 4 November 2018. [Online]. Available: https://www.techrepublic.com/article/microsoft-may-be-the-worlds-largest-open-source-contributor-but-developers-dont-yet-care/.

[44] Q. Jiang, J. Qin and L. Kang, "A Literature Review for Open Source Software Studies," vol. 9191, pp. 699-707, 2015.

[45] A. Kubow, "A Business Case for Open Source – Why You Should Contribute to the Open Source Community," freeCodeCamp, 14 September 2021. [Online]. Available: https://www.freecodecamp.org/news/a-business-case-for-open-source/#:~:text=Contributing%20to%20Open%20Source%20Attracts,the%20upgraded%20or%20scaled%20versions.

[46] Z. Staff, "THE BENEFITS OF OPEN SOURCE SOFTWARE," Zivtech, 12 July 2019. [Online]. Available: https://www.zivtech.com/blog/benefits-open-source-software.

[47] S. Phipps, "License compliance is not a problem for open source users," Opensource.com, 25 October 2010. [Online]. Available: https://opensource.com/law/10/10/license-compliance-not-problem-open-source-users. [Accessed 11 March 2022].

[48] E. N. Hahn, "An Overview of Open-Source Software Licenses and the Value of Open-Source Software to Public Health Initiatives," vol. 32, no. 4, pp. 690-698, 2014.

[49] R. J. Fakhlina and D. F. Saputra, "Development and Contribution of Open Source Software Communities for the Library Progress in Indonesia," vol. 5, no. 2, pp. 150-159, 2019.

[50] R. SASS, "Top 10 GPL License Questions Answered," WhiteSource, 2 June 2021. [Online]. Available: https://www.whitesourcesoftware.com/resources/blog/top-10-gpl-license-questions-answered/.

[51] "What is the MIT License? Top 10 questions answered," snyk, [Online]. Available: https://snyk.io/learn/what-is-mit-license/.

[52] "What is the BSD License? Top 10 questions answered," snyk, [Online]. Available: https://snyk.io/learn/what-is-bsd-license/.

[53] R. SASS, "Top 10 Apache License Questions Answered," WhiteSource, 6 May 2021. [Online]. Available: https://www.whitesourcesoftware.com/resources/blog/top-10-apache-license-questions-answered/.

[54] T. J. Lehman and A. Sharma, "Software development as a service: Agile experiences," *2011 Annual SRII Global Conference,* pp. 749-758, 2011.

[55] S. Barjtya, A. Sharma and U. Rani, "A detailed study of Software Development Life Cycle (SDLC) Models," vol. 6, no. 7, pp. 22097-22100, 2017.

[56] J. Kwon, "What is Software Development Lifecycle (SDLC)?," 9 May 2019. [Online]. Available: https://joycekwon.medium.com/what-is-software-development-lifecycle-sdlc-523fd09340a6.

[57] R. D. Amlani, "Advantages and Limitations of Different SDLC Models," *International Journal of Computer Applications & Information Technology,* vol. I, no. III, 2012.

[58] S. Kumar and P. Dubey, "SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) ANALYTICAL COMPARISON AND SURVEY ON TRADITIONAL AND AGILE METHODOLOGY," vol. 2, no. 8, pp. 22-30, 2013.

[59] "Principles behind the Agile Manifesto," Agile Manifesto, 2001. [Online]. Available: https://agilemanifesto.org/iso/en/principles.html.

[60] B. V. d. M. Carvalho and C. H. Pereira, "Scrum agile product development method - literature review, analysis and classification," vol. 9, no. 1, pp. 39-49, 2011.

[61] M. Sameen Mirza and S. Datta, "Strengths and Weakness of Traditional and Agile Processes --- A Systematic Review," vol. 14, no. 5, pp. 209-219, 2019.

[62] "Agile tools for software teams," ATLASSIAN, [Online]. Available: https://www.atlassian.com/software/jira/agile#scrum. [Accessed 7 May 2022].

[63] K. Siau, "An Analysis of Unified Modeling Language (UML) Graphical Constructs Based on BWW Ontology," *Journal of Database Management,* vol. 21, 2010.

[64] A. Suryana, "UNIFIED MODELING LANGUAGE," *Academia.*

[65] Lucid, "What is a Flowchart," Lucid, [Online]. Available: https://www.lucidchart.com/pages/what-is-a-flowchart-tutorial. [Accessed 15 March 2022].

[66] N. Hebb, "Flowchart Symbols Defined," BreezeTree Software, [Online]. Available: https://www.breezetree.com/articles/excel-flowchart-shapes/. [Accessed 30 June 2022].

[67] A. Suprunenko, "How To Succeed In Wireframe Design," 29 April 2020. [Online]. Available: https://www.smashingmagazine.com/2020/04/wireframe-design-success/. [Accessed 29 June 2022].

[68] anuupadhyay, "5 Simple Steps to Create Wireframe in Software Design," GeeksforGeeks, 22 March 2021. [Online]. Available: https://www.geeksforgeeks.org/5-simple-steps-to-create-wireframe-in-software-design/. [Accessed 29 June 2022].

[69] "Sitemaps," Backlinko, [Online]. Available: https://backlinko.com/hub/seo/sitemaps. [Accessed 30 June 2022].

[70] "Learn about sitemaps," Google Search Central, [Online]. Available: https://developers.google.com/search/docs/advanced/sitemaps/overview. [Accessed 30 June 2022].

[71] Lynx Information Systems, "About Lynx Information Systems," [Online]. Available: http://www.lynxinfo.co.uk/about.html . [Accessed 30 March 2022].

[72] Interactive Network Technologies, "INT Visualization Product Ecosystem," [Online]. Available: https://www.int.com/products/. [Accessed 30 March 2022].

[73] Interactive Network Technologies , "What Is IVAAP? [Overview]," YouTube, 25 July 2020. [Online]. Available: https://www.youtube.com/watch?v=UfSsktl7RaQ. [Accessed 31 March 2022].

[74] Interactive Network Technologies, "IVAAP DEMO: MAP INTERFACE AND WELL LOGS," [Online]. Available: https://www.int.com/ivaap-demos/map-interface-and-well-logs/. [Accessed 2 April 2022].

[75] Schlumberger, "Who We Are," [Online]. Available: https://www.slb.com/who-we-are. [Accessed 31 March 2022].

[76] Schlumberger, ProSource Front Office, Schlumberger, 2013.

[77] C. Chapman, "Why Use Material Design? Weighing the Pros and Cons," [Online]. Available: https://www.toptal.com/designers/ui/why-use-material-design. [Accessed 1 April 2022].

[78] N. Babich, "Flat vs. Material vs. Skeuomorphic Design Examples," Adobe, 24 November 2020. [Online]. Available: https://xd.adobe.com/ideas/principles/web-design/flat-vs-material-skeuomorphic-examples/. [Accessed 1 April 2022].

[79] AsoftClick, "Material Design Apps for Android," 10 October 2021. [Online]. Available: https://asoftclick.com/best-material-design-apps/. [Accessed 6 April 2022].

[80] Geoapify, "Leaflet vs OpenLayers. What to choose?," 12 April 2019. [Online]. Available: https://www.geoapify.com/leaflet-vs-openlayers. [Accessed 2 April 2022].

[81] LYNX Information Systems, "Lynx Software - Licence Pricing," LYNX Information Systems, [Online]. Available: http://www.lynxinfo.co.uk/download-pricing.html.

[82] INTViewer, "INTViewer Rapid Geoscience QA/QC," INTViewer, [Online]. Available: https://www.int.com/products/intviewer/.

[83] G. Balinov, "Traditional Web Design vs. Growth-Driven Design," HYPE Digital Agency, [Online]. Available: https://agencyhype.com/blog/traditional-web-design-vs-growth-driven-design.

[84] J. M. Carroll, "2. Human Computer Interaction - brief intro," in *The Encyclopedia of Human-Computer Interaction, 2nd Ed.*, pp. 21-62.

[85] R. Valverde, Principles of Human Computer Interaction Design, Lambert Academic Publishing, 2011.

[86] E. Wong, "Shneiderman's Eight Golden Rules Will Help You Design Better Interfaces," Interaction Design Foundation, 2021. [Online]. Available: https://www.interaction-design.org/literature/article/shneiderman-s-eight-golden-rules-will-help-you-design-better-interfaces.

[87] N. Honest, "Role of Testing in Software Development Life Cycle," *International Journal of Computer Sciences and Engineering,* vol. 7, no. 5, pp. 886-889, 2019.

[88] T. Jindal, "Importance of Testing in SDLC," *International Journal of Engineering and Applied Computer Science (IJEACS),* vol. 1, no. 2, pp. 54-56, 2016.

[89] B. Kumari, N. Chauhan and H. H. Syed, "A COMPARISON BETWEEN MANUAL TESTING AND AUTOMATED TESTING," *Journal of Emerging Technologies and Innovative Research,* vol. 5, no. 12, pp. 323-331, 2018.

[90] Vue.js, "Testing," Vue.js, [Online]. Available: https://v2.vuejs.org/v2/guide/testing.html. [Accessed 16 March 2022].

APPENDICES

Flowchart

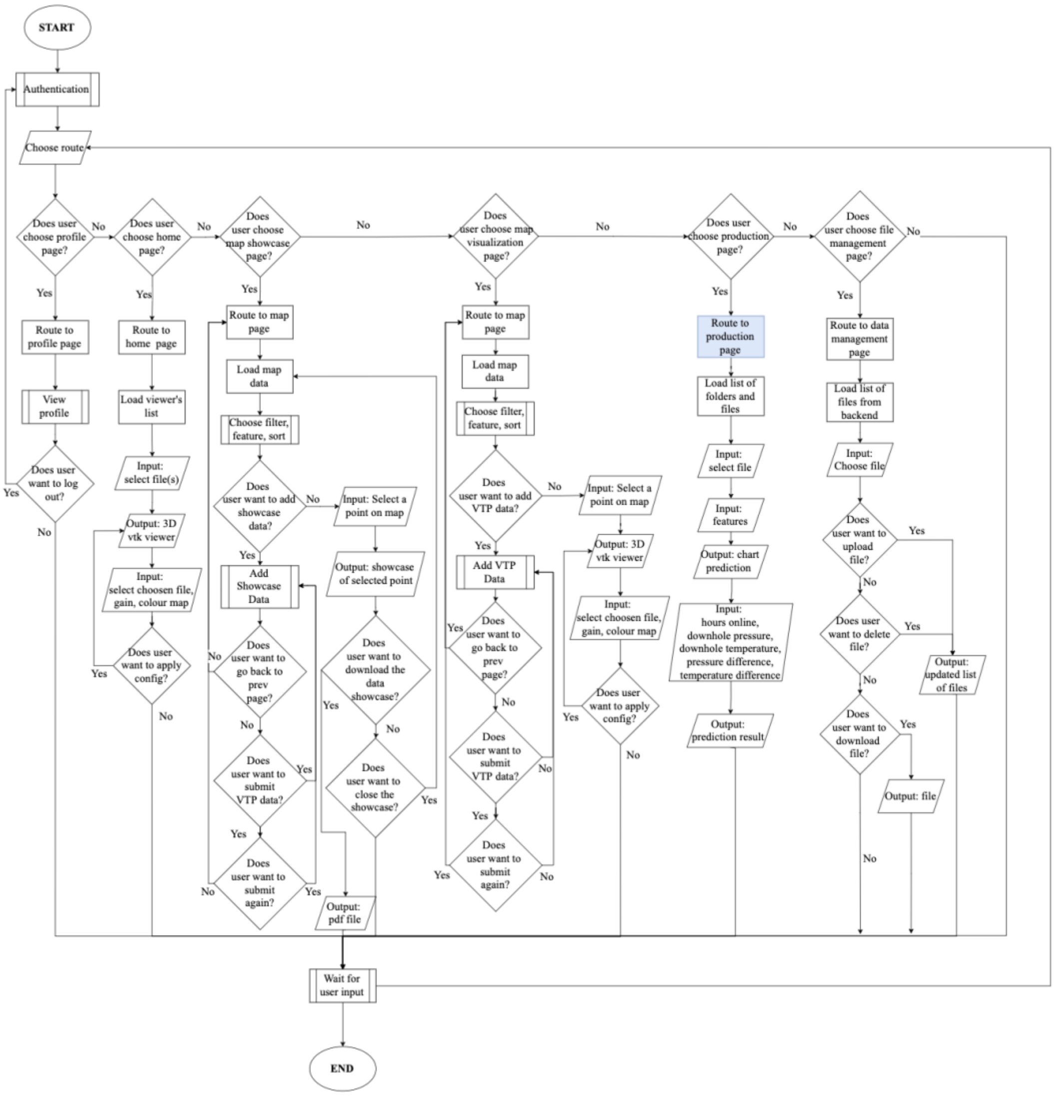


Figure A.0.1 Full Flowchart

Scope of Work

VDR Website Application

The party involved in this development consist of:

Product Owner: **Ardimas Andi Purwita**

Client: **Widodo Nugroho - PT Geodwipa Teknika Nusantara (GTN)**

Developers: **Chan Elizabeth**

Kotrakona Harinatha Sreeya Reddy

Vicky Vanessa

Problem Statement:

The client wishes to have a more cost-effective application as an alternative since the application they were using is not cost-effective enough for the client's company.

Scope of work of the project:

The developers are to create a website application frontend prototype for the VDR Website Application to Visualize Production Data for the Oil and Gas Industry. The aims of a fully functional prototype are:

- visualizing oil and gas data,
- being more cost-effective, and
- a predictive model capable of predicting oil and gas production.

The prototype will consist of the features that the client demand which are:

1. Visualization of oil and gas data

   Building a viewer's page which allows the user to choose which file they choose to visualize. Visualization is available in 2D and 3D images, i.e., well logs and seismic data. The file they choose is the file that the user uploaded in the file management, which will be described later.
2. Map application along with the showcase feature

   An application which allows the user to see and upload the location of oil and gas reserves. It shows the data the client has for the reserves for the showcase, e.g., tabular data, snapshots of the location.
3. Prediction model to predict the oil and gas production

   Using pressure and temperature sensor data to predict oil and gas production values so that the user can focus on wells that contain more oil and gas.
4. File management to store the user's files

   The user can store their files and store them into folders. These files can and will be used in the other features.

Approved by

Ardimas Andi Purwita
Product Owner

Widodo Nugroho
Client

Figure A.2 Proof of Scope of Work