Table of Contents

List of Figures

List of Tables

CHAPTER 1
INTRODUCTION

This chapter clarifies the project that the author worked on, including the background story and problem that motivate the start of the project. Moreover, the scopes of work and objectives of this project are discussed in this chapter.

1.1 Background

In the 19th century, the premodern era, oil and gas were mainly used as a source of lubricant, medical use, source of light, and energy source [1]. At present, oil and gas have been the center of modern industrial society and the major geopolitical objective and majority of energy sources for nations for centuries [2]. Both crude oil and natural gas are used as the source and material of extensive types of chemicals and materials, they are the majority and dominant source of food production, pesticides, and transportation fuels [1].

Several countries, including Indonesia, are greatly reliant on the use of oil and gas for the growth of the national economy as well as the industry [3]. It is one of Indonesia's economic pillars, contributing 21.09% to Indonesia's gross domestic product (GDP) from oil exports alone in 2004 according to the Organization of Petroleum Exporting Countries (OPEC) [2]. Pertamina, which is the state-owned oil-and-gas industries is involved in the government's oil and gas exploration and production activities as well as help improve the Indonesian GDP by contributing to more than 60% of Indonesia's

GDP [3]. Furthermore, they also provide more quality products to meet their consumer demands [3].

In order to help the oil and gas industry to be more productive, there are many existing dashboard-based software applications, such as, Virtual Data Room (VDR). VDR is an online archive of oil and gas resources [4]. A VDR can help the user to analyze their oil and gas production data. These applications include Lynx and INTviewer [5] [6]. INTviewer is a platform that offers users the ability to check their seismic data, geospatial integrity, and process their datasets, which could cost up to $4,000/person a year [5]. On the other hand, Lynx is a platform that offers access to geophysical and geographical information systems (GIS) while also offering the 2D and 3D seismic viewers that cost at least €250 a year for each user [6].

However, while this software is useful and benefits the oil and gas industries, the expenses needed to be able to use the software are expensive. Thus, the author's team client, PT. Geodwipa Teknika Nusantara [7], wishes to have an application with customized features at a cheaper price. The product owner, Mr Ardimas Andi Purwita, also states that there are several oil and gas companies, including the author's team client, that wants a cheaper and customized application [8]. Therefore, the main goal of this project is to develop a VDR that has customized features at a lower cost using open-source libraries. This thesis will be focusing on the frontend development of the VDR website application, Section 1.2.2 will delve into the author's role in this project.

1.2 Scope

The scope of the problem, especially the solution that the author team thought of, and the scope of work and responsibility that the author had, and the task the author team had as a group are described in this chapter.

1.2.1 Group Scope of Work

The main problem of this project is the high cost of the software with preferable and useful features. To solve the problem, which was to develop a VDR website application with various features at a lower cost, the solution that the author discussed with their team was to make use of open-source libraries along with hand-picking essential features from the existing software stated previously to be used as well as developing custom features requested the author's customer.

The VDR website application's features are based on similar existing software applications, Schlumberger's ProSource Front Office [8]. The website application referenced the data visualization support of ProSource Front Office in the form of lines, bars, bubbles, and pie charts and their integration with ArcGIS Arcmap to perform their geospatial analysis [9]. Aside from visualizing, the website application will consist of the following features for the user, such as:

- upload and store the files,
- delete and download the files,
- viewing map, and
- viewing data using a predictive model.

The responsibility as a team for this project was to create the different kinds of visualizations and implement the various features and functions the author's team had made to the VDR website application. The members of the team had different responsibilities and importance to successfully finish this project as presented in Table 1.1.

Table 1.1 Scope of Work of the Team

Name	Thesis Title	Role
Vicky Vanessa	VDR Website Application to Visualize Production Data for Oil and Gas Industries – Frontend Development	Designing the UI of the front end of the website application, visualizing the data of oil and gas, along with being the Scrum master.
Elizabeth Chan	VDR Website Application to Visualize Production Data for Oil and Gas Industries – GIS Application	Design the frontend of the proposed VDR application that uses GIS, as well as the testing and act as Scrum team.
Kotrakona Harinatha Sreeya Reddy	VDR Website Application to Visualize Production Data for Oil and Gas Industries – Predicting Oil and Gas Production Data by Using Data Science	Collecting and processing data, and using them to develop predictive models, visualizing them through diagrams such as charts while doing the data analytics and act as Scrum team.

1.2.2 Individual Scope of Work

The author's responsibility in this VDR website application project was the frontend of the website, specifically on the design and user interfaces (UI), and the logic of the frontend development. In addition, the author is in charge of making one of the

visualizations, the 2D and 3D visualization using chart, for the oil and gas data. Moreover, the author is responsible for creating the UI of these pages:

- viewer's page,
- file managements page,
- production's page,
- user profile's page, and
- administrator's page.

1.3 Aims and Benefits

This section explains the aims and benefits of developing this website application.

1.3.1 Aims

Apart from solving the problem stated in the previous section, the primary aim of developing this VDR website application is to help the oil and gas industry by:

- building a cheaper VDR website application that provides custom features,
- visualizing oil and gas volume reserve resources, and
- building a predictive model capable of predicting oil and gas production.

Aside from the aim of the VDR website application in general, the author's aim, based on Table 1.1, is to design the frontend of the VDR website application. In the frontend, the author aims to build and design the visualization of oil and gas data, and the file management.

1.3.2 Benefits

Respective to the aims, the benefits of building the VDR website application are to:

- increase the use of local services in the oil and gas industries,
- developing a high quality and affordable VDR website application, and
- helping users comprehend complex oil and gas data.

Some local companies would like to have their own VDR that match their needs. Aside from that, the author team wishes for the VDR website application to be known internationally. Apart from providing advantages for the oil and gas industries, the application is also useful to aid the engineers in comprehending sophisticated data and to gain a better understanding of how the data could be used.

1.4 Structure

This thesis consists of 5 chapters as defined further in this section

Chapter 1: Introduction

In Chapter 1, the topic is introduced along with the description of the author's scope, objective, and aim for this project.

Chapter 2: Theoretical Foundation

In Chapter 2, the project theories, why it is carried out in a certain way, how the project will be made, as well as what it is the author used to develop the project is described and explained.

Chapter 3: Problem Analysis

In Chapter 3, the VDR website application will be compared to similar existing applications. Furthermore, the solution to the problem described previously will be discussed, especially the reason on why the author choose to use a particular tools or frameworks.

Chapter 4: Solution Design

In Chapter 4, the features and functions will be explained, along with the flow of the system for the user. In addition, the design of the website application and planning on how the testing will be performed will be further discussed.

Chapter 5: Implementation

In Chapter 5, how the system will be implemented and put together in developing the VDR website application.

Chapter 6: Discussion

In chapter 6 will be having a discussion regarding the project.

Chapter 7: Conclusion

In Chapter 7, the final chapter, will conclude at the end of the project.

CHAPTER 2
THEORETICAL FOUNDATION

This chapter discusses the theories that the author uses to develop and design this project. It highlights how the author and the team design the project together, what theory and concept they use to aid in building the project. Aside from that, this chapter also describes the reasoning behind the theory thoroughly throughout the chapter. Starting from the concept of VDR website application, OSS, Software Development Life Cycle (SDLC) to testing. Specifically on the approach, sources, limitations, and the frameworks of the frontend and the development that the author uses to build the project.

2.1 Virtual Data Room

Data room is a promising valuable asset of intelligence about oil and gas resources that are available for sale, undoubtedly used by the oil and gas companies who wish to distribute their share of assets [4]. Interested buyers' Mergers and Acquisitions (M&A) team would visit the data room to analyze and inspect the data placed and arranged by the seller, then decide whether or not the assets are worth purchasing [4]. Data rooms, with an abundant amount of confidential oil and gas information sources, have different forms, the Physical Data Room (PDR), VDR, or both combined [4]. Visiting a data room depends on the seller, whether or not they allow it. Most often than not, they will not allow a second look once they decided to close it [4]. Thus, both sellers and buyers must be conscientious assure everything is going well [4].

PDR is a closely monitored room to ensure the secrecy of the data placed by the seller or their representatives [4]. It has a lot of disadvantages as it is expensive, time-consuming, as well as an inconvenient geographical location [4]. VDR is built as an alternative to the cumbersome way of PDR. It also has replaced the majority of PDR [4]. VDR is a controlled-access website that allows loads of documents to be uploaded for a limitless amount of users to analyze documents [4]. VDR is available and accessible continuously non-stop and, in some circumstances, should be easy and quick to set up [4]. Any information on the VDR could be updated, modified, or deleted at any time [4].

VDR has a lot of benefits over the traditional PDR for the clients [4]. VDR offers information and documents at a faster speed, lower fees, as well as better efficiency [4]. Moreover, since the platform is online, it allows access no matter the current location and region of the clients [4]. Aside from that, by using VDR, clients do not have to go their way to send their teams to an inconvenient location of a PDR [4]. At times, it could be complicated for the clients to send the right team to the PDR, but the existence of VDR saves that complication [4]. Based on Schlumberger, there are three possible features of VDR which are:

- full series of petrotechnical solutions,
- generate and download reports as well as other documentation, and
- available to access from anywhere around the world regardless of OS and devices [10].

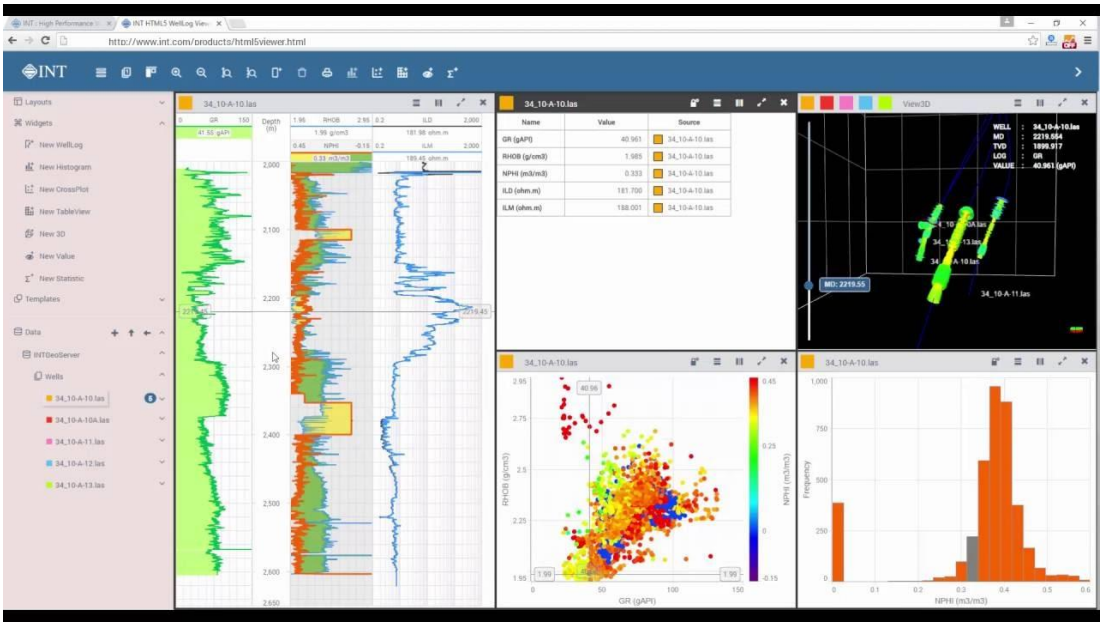


Figure 2.1 Dashboard in Virtual Data Room

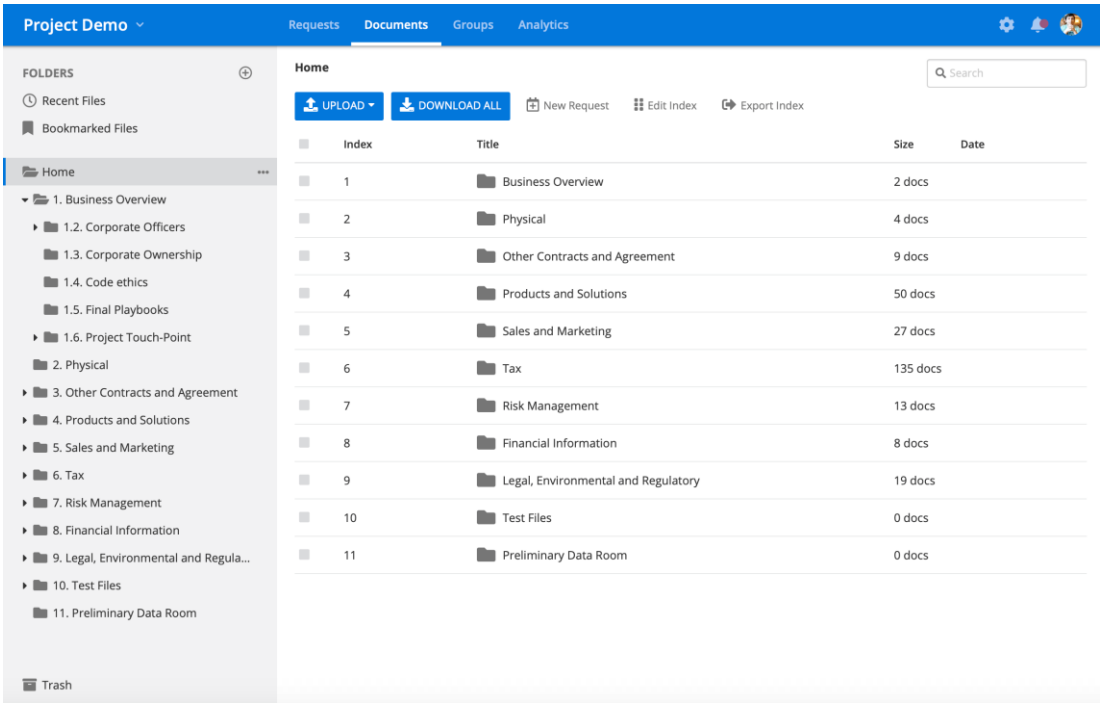


Figure 2.2 File Management in Virtual Data Room [11]

Figure 2.1 and Figure 2.2 show examples of some of the features available in a VDR. Figure 2.1 shows an example of a data visualization of oil and gas data, while Figure 2.2 shows the file management features in a VDR.

2.1.1 vtkPolyData

Visualization Toolkit or VTK Polydata (VTP) is a file format for storing VTK model [12]. VTK is used to process and create 3D computer graphics [13].

2.2 Website Application

VDR will be developed in the form of a website application. The website uses the World Wide Web (WWW), which was introduced to assist in achieving access to information from any source in a consistent and convenient method [14]. Introduced in the 1990s by the Conseil Européen pour la Recherche Nucléaire (CERN) in order to accommodate the enormous amount of data and documents required to be shared with other scientists [14]. The method adopted is by using HyperText Transfer Protocol (HTTP) which was designed to allow one computer to request data and documents from the server computer [14]. This helps users to access the documents through the computer [14]. WWW was perceived as a large archive of information that allows access to numerous amount of users [14]. Website application refers to the program that could return the page of data from the server to return its result to the user, some describe it as a combination of hypermedia and information system [14]. At present, website application is a powerful platform that supports individuals, SMEs, communities of users, and large corporate businesses [14]. In 1999, Fratenali stated the requirements of a website application which are:

- the need to handle both structured and non-structured data, referring to the database record and multimedia items, respectively,
- navigational interface in order to assist exploratory access,
- impressive graphical quality, and

- customizable and the dynamic adaptive content structure, navigation, and presentation style [14].

A website is a combination of connected and related website pages which occupied on a single server are available for a variety of categories and usages [14]. It continues to rapidly evolve while maintaining the basic concepts of WWW which are Uniform Resource Application (URL), HyperText Markup Language (HTML), and HTTP [14]. URL is a naming system that prescribes the way objects could be identified and located as well as searchable based on their attributes or names [14]. HTML is a document on the web in which the content can be displayed and linked to other documents to achieve enhanced visual presentation [14]. Meanwhile, HTTP is a communication of a request and reply protocol that contains the most used basic operations of GET, PUSH, PUT, DELETE which could be requested to the URL [14]. Website applications use a similar concept to that of software, they have a UI, may contain and manage a vast amount of data on the server, as well as the Model View Controller (MVC) design pattern [14]. In fact, several frameworks are developed in support of the design and development of the MVC website application [14].

Aside from that, research on modern website applications pointed out that there are several standard features in website applications which are:

- search,
- tagging,
- user participation, and
- UI and collaboration [14].

Searching and tagging have been a mechanism to assist users in finding the information that they need amongst the abundance amount of information on a website [14]. User participation is to create and design the structure of the data [14]. While UI and collaboration primarily exist in a messaging system, for instance, Google Mail and Skype, which would notify the user in real-time when someone in their contact is online [14].

In a website application, a three-tiered architecture model is widely used. It is an architecture with three different levels, as presented in Figure 2.3 [15]. The principle of this architecture model is that business logic is the vital core of the entire application system [15]. According to Figure 2.3, if the process is modified, the tiers would need to be changed correspondingly [15]. Similarly, if the data structure in the storage is changed, every tier would be changed to adjust to it [15]. As mentioned in Section 1.2, the frontend development is part of the scope. Thus, Section 2.2.1 will delve further into the presentation tier of this architecture.

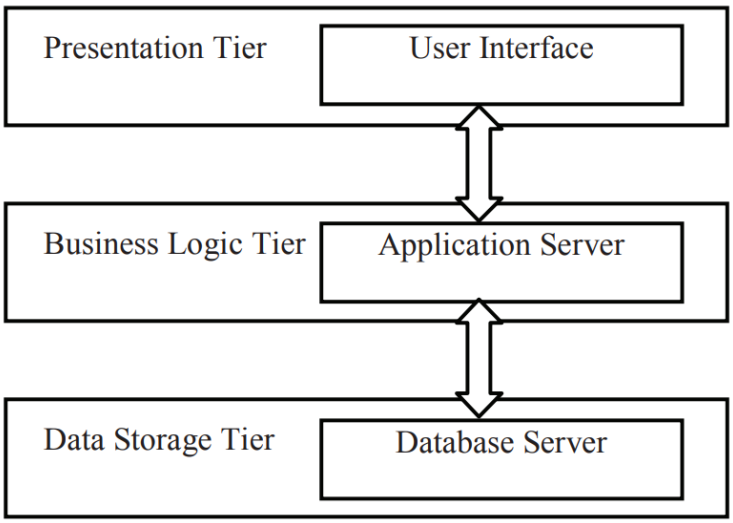


Figure 2.3 Three-tiered Architecture Model

2.2.1 Frontend Development

Website technology has expanded over the years along with the development of the internet, especially the HTML that leads the frontend development [16]. Frontend development has evolved simultaneously with the demand of business needs [17]. The traditional model which used JQuery was not enough to fully develop the website frontend that meets the interest of the current era of website development [17]. JQuery is an open-source JavaScript library to help the handling of HTML Document Object Model (DOM), ajax, and Cascading Style Sheet (CSS) [17]. Although it is still widely used in website development, more complex libraries that fit the business needs, for instance, Single Page Application (SPA) and Progressive Web Application (PWA) [17]. Developers frequently and mainly use frontend frameworks when developing a large and sophisticated website application, it allows them to reuse repetitive code, saving time in designing and developing the application [18]. There are numerous JavaScript frameworks and libraries for SPA, among those, the most popular frontend frameworks are Angular, React, and Vue [16] [19]. SPA is a website interface consisting of individual components that can be easily updated or replaced individually [18] [19]. Thus, preventing a reload of the page for every action the user made while loading dynamically [18] [19].

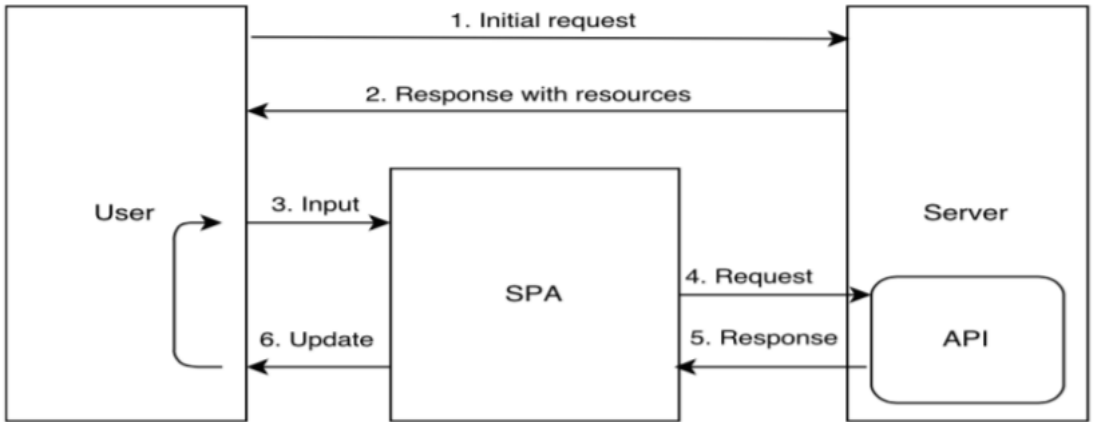


Figure 2.4 Communication between the user and a web browser with Single Page Application [16]

Figure 2.4 presents the communication between the user, SPA, and the backend server. The initial request is sent through HTTP by the user's web browser [18]. The request sent to the backend server is through JavaScript Object Notation (JSON), the server will send back JSON data in response [18]. As a result of the communication process described, SPA has a quick response/render time for the user [18]. Moreover, for the developer, it is advantageous for them as the presentation side and logic are separated, allowing them to be able to work and update on them separately [18]. Aside from that, the data transmission, sending, and receiving to and from the server are faster [18].

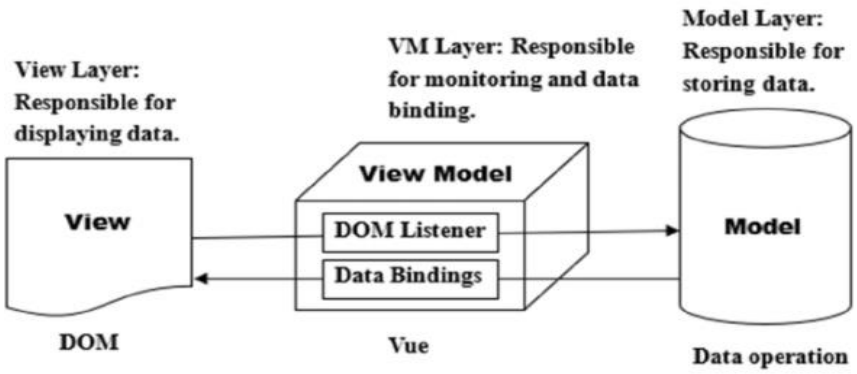


Figure 2.5 Model View View-Model Architecture Diagram of Vue [15]

As mentioned previously, Vue is one of the popular open-source JavaScript ES6 frontend frameworks [18]. Creating a SPA website with comprehensive and high-performance in developing an interactive interface [18]. Vue uses Model View View-Model (MVVM) design pattern that supports its data-driving and component-based development [17]. MVVM architecture is unique from the conventional operation of DOM nodes, as shown in Figure 2.5 [17]. Vue is more focused on the view-model aspect of MVVM, being bonded to both directions responsively [17]. The bindings

made Vue respond asynchronously to updates from the data of the user for the DOM to be synchronized with the updates [17] [20].

Vue is an excellent choice as a frontend framework for the reason that it has significant advantages over other frontend frameworks such as:

- greater efficiency, usability, user experience, fast and light,
- rendering broad modular models without additional time,
- great reactivity between HTML and JavaScript code, and
- DOM is modified accordingly to any updates from the user [19].

All these advantages are what makes it great as a SPA, furthermore, Vue is adaptable to the user, usable for users with poor connection, as well as great for computers with insufficient resources [19].

2.2.2 User Interface

UI is the interface of an application or a system focusing on the design as well as a moderator between a user and a system [21] [22]. UI is crucial in terms of increasing the usability of an application [21]. There are several forms of UI which are:

- GUI,
- Voice-controlled Interfaces (VUI), and
- gesture-based interfaces [2].

GUI is a user interaction through visual representation on a digital control panel, such as a computer or a mobile phone [22]. VUI is user interaction with their voice, most smart assistants use this, for instance, Siri and Alexa [22]. While gesture-based interfaces use body motion in a 3D space, for example, a virtual reality (VR) [22].

As UI designers, their task is to create the appearance and layout of an application [23]. It is their responsibility to design an attractive and themed UI according to the application purpose [23]. A UI designer is essentially different from a UX designer [23]. While a UI designer designs the looks of a UI, UX designers focus on how the elements in the interface work [23]. There are several UI design guidelines which are recommendations for designers to follow when creating a UI for applications [24]. These guidelines consist of:

- basic design principles derived from research,
- standards, and
- style guide [24].

The design principles could consist of design methodology and concept [24]. Some companies and organizations have their own concept of UI which is documented for UI designers in their company to use [24]. In addition, a number of the documents are also published as official guidelines [24].

2.2.3 Material Design

Material is an adaptable system consisting of guidelines, components, as well as tools to support a good practice of UI design made by Google [25]. Material components are available for Android, iOS, web, and Flutter to help developers develop their UI in accordance with the guidelines provided by Material [25] [26]. Material offers customizable UI components, enabling developers to build an appealing and functional UX [25] [26]. It is inspired by the real physical world and its material and textures, for instance, how light is reflected and cast their shadow, specifically using paper and ink as the medium [25]. Material components cover a range of the essentials of UI, for instance:

- display,
- navigation,
- actions,
- input, and
- communication [25].

Material Design in the web is available on SPA frameworks, such as Angular Material for Angular, MUI for React, as well as Vuetify for Vue.

Vuetify is a UI framework built for Vuejs, and unlike Material Design on other frameworks, Vuetify is designed from scratch to make it easy for learning [27]. It has active development as it is being patched frequently, seeking out issues in the community, having more bug fixes, and got enhanced often [27]. Vuetify has a massive community to support developers and collaborate with them [27].

2.3 Application Program Interface

An Application Program Interface or API enables the sharing of data and functionality to a third party [28]. This provides services and products to share their data and functionality through an interface [28]. API is an intermediary layer which transfer data between an application and a web server [28]. Steps of how an API work are:

1. a client application requesting from an API,
2. API calls the program after receiving a valid request,
3. the program sends a response back to the API, and
4. the API transfer the response to the client application [28].

A request from an application is through a Uniform Resource Identifier (URL), a request verb, header, sometimes including a request body [28]. APIs are created to be

used by a computer or a system [28]. API offers security as it may include authorization credentials as it is positioned as a middleman between two systems [28]. API comply to REST principle along with JSON formatting and usually built for HTTP [28].

2.4 JavaScript Object Notation

JSON is a lightweight format for dealing with data [29] [30]. Usually used to send data from a server to a web page [29]. JSON is a language independent text format but uses conventions which is familiar to programmers to languages such as C, Java, JavaScript, and Python [30]. JSON is created based on two structures, which are:

- a collection of value-pair data such as object, dictionary, or hash table, and
- a list of values for instance, array or vector [30].

These structures are universal data structure which format could be used interchangeably with programming languages [30].

2.5 Authentication and Authorization

Authentication and authorization are used for security which permits safety of a system [31]. Although both terms are related to each other, they are distinctly a different from one another [31]. The identity of the user is verified by authentication, whereas their authorities are verified by authorization [31] [32]. In a process, authentication is done first prior to authorization [31] [32]. The authorization usually through the exchange of information according to an access token by the Open Authorization (OAuth) 2.0 framework [32]. Access token is a credential that could be used by a system to access an API [33]. The access token will communicate with the API that the bearer has been authorized to perform the actions that has been granted to them [33]. There are 2

popular formats of an access token, which are through an opaque string and JSON Web Tokens (JWT) [33].

Other than OAuth 2.0, there are other protocols that could be used for user authentication which are [34]:

- kerberos,
- Lightweight Directory Access Protocol (LDAP), and
- Security Assertion Markup Language (SAML).

Kerberos aids in network authentication used to validate client or server during a network using cryptographic key [34]. LDAP is to decide an individual, organization, or a device during a network regardless of being in a public or corporate internet [34]. SAML is an Extensible Markup Language (XML) based authentication data format which provides authorization between the identity provider and the service provider [34].

OAuth, or OAuth2 specifically is an HTTP based protocol for authorization of applications to use user information [35]. OAuth authorize a third-party application to access user related information for instance username, full name, and date of birth, from applications like Google or Facebook without giving the third-party applications any sensitive information such as the user's account password [35]. An example of an application with OAuth would be when a login page of an application has an option to log in with their Google account or Facebook account as depicted in Figure 2.6 [35]. The method of permitting authorization to a third party is by granting access tokens from a authorization server with the user's permission [35].

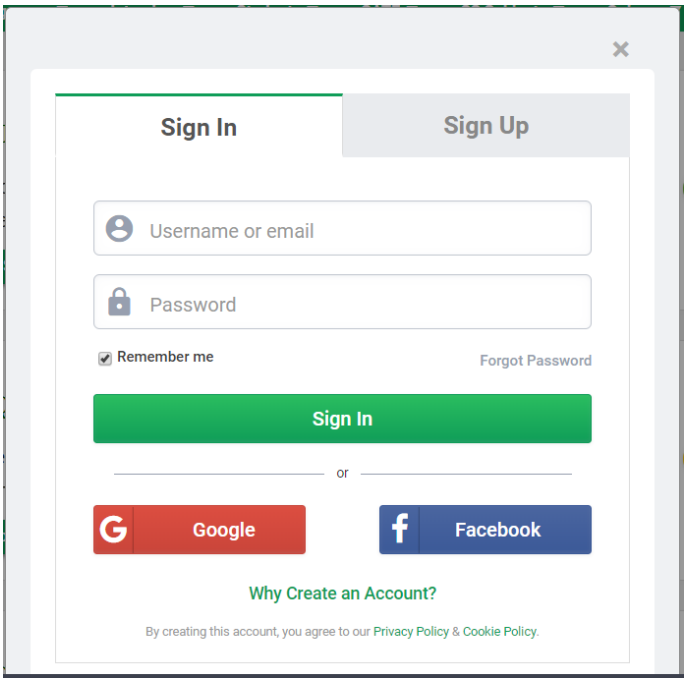


Figure 2.6 OAuth Login Page [39]

In OAuth mechanism, there are three components which are:

1. OAuth provider,
2. OAuth client, and
3. owner [35].

OAuth provider is as stated in the previous examples, logging in with Google account or a Facebook account [35]. The client is the application which the user authenticated to use their information [35]. While the owner is the user who authenticate the sharing of their own information during login [35].

OAuth2 supports different type of grants or flows which are ways for applications to retrieve their access tokens, such as:

- client credential flow,
- authorization code flow,

- resource owner password flow, and
- implicit flow [36].

These flows are used depending on the application type along with other things to consider [36]. If the client is a resource owner, then the developers could consider using client credential flow [36]. The resource owner is the entity in which grant an access to a protected system [36]. Assuming that the client is a regular website application running on a server, authorization code flow could be considered [36]. Supposed that the client is trusted with user credentials, resource owner password flow may be considered [36]. With the condition that the client is an SPA but does not need an access token, they could take implicit flow into account [36]. The client referred in the statement beforehand refers to the application requesting an access to a protected system [36].

2.5.1 Resource Owner Password Flow

Resource Owner Password (ROP) flow is a type of flow which requires the user to provide their credentials which would be their username and password [37]. It is necessary for the user to trust the application as the credentials are sent and verified in the backend and could be stored [37]. Figure 2.7 illustrates the process of a ROP flow. After the user logs in with their credential, the OAuth will validate the user's credential and return the access token [37]. The access token would be later required when requesting from the API endpoints [37].

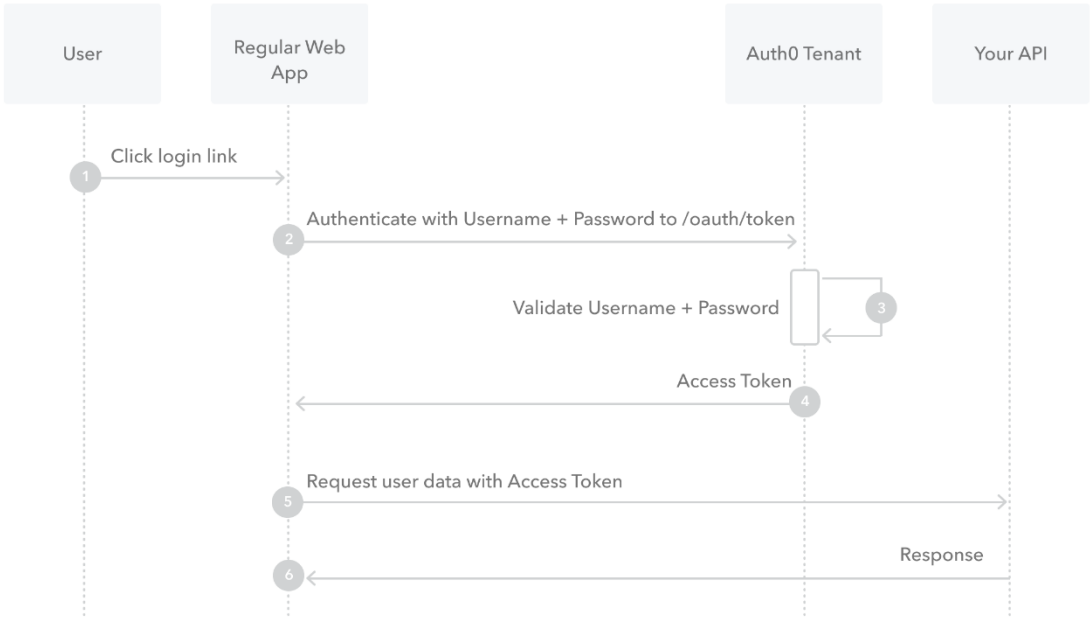


Figure 2.7 Resource Owner Password Flow [41]

2.6 Encryption

Encryption is a significant tool in preserving privacy of data which information is comprehensible to the authorized party [38] [39]. It conceals the information in form of random scramble of data or translating the data into an incomprehensible string [38]. This scramble of data is known as ciphertext [38]. Figure 2.8 show an example of how a human-readable information turns into a ciphertext [38]. An encryption requires a cryptographic key which both the sender and the receiver assent to [38]. The receiver could decrypt the ciphertext back into a plain text using the cryptographic key [38].

Figure 2.8 Encryption Example

There are two different types of encryptions which are symmetric encryption and asymmetric encryption [38]. Symmetric encryption requires a secret key for encryption and decryption [38] [39]. Algorithms with symmetric encryption are:

- Data Encryption Standard (DES),
- Advanced Encryption Standard (AES), and
- SNOW [38] [39].

Whereas asymmetric encryption is known as public key encryption, which require two keys [38] [39]. A public key is required for encryption while a private key is for decryption [38]. Algorithms with asymmetric encryption are:

- Rivest-Shamir-Adleman (RSA),
- ElGamal, and
- Cramer-Shoup [38] [39].

2.7 Open Source Software

VueJs framework and the libraries available for Vue is part of the OSS. OSS is a user-driven, self-organizing team of contributors formed through online interaction, who develops libraries and codes that are available under a license that could be reused, modified, improved by the communities [40] [41] [42]. Over the emanating development over the last decades, OSS has been becoming important as it is valuable to significant information technology (IT) companies [40]. For instance, Google created a platform to publicly host OSS projects for the community in 2005 named Google Code [40]. Figure 2.9 presents the difference between proprietary software with open source code and closed source code. While the corporate or individual could choose to publicize the code, according to Figure 2.9, software with open source could

offer more flexibility and adaptability as it is in development by the community as it could be modified and fixed repeatedly [40].

Both users and developers are perchance licensed to use and modify the code and distribute the modification and improvements they made at their discretion [40]. However, according to Open Source Initiative (OSI) and Free Software Foundation (FSF), the free source of software does not entail that it is free of cost [40]. Instead, it

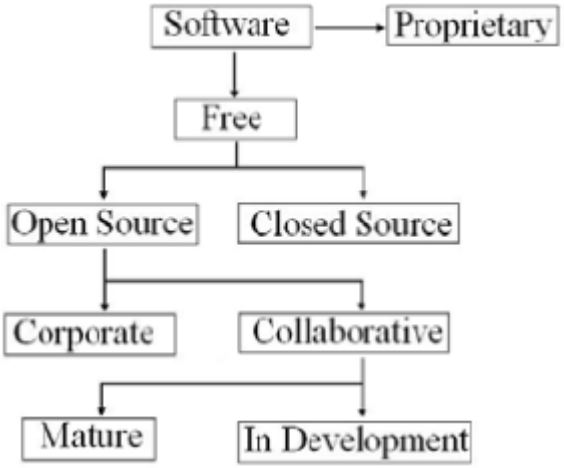


Figure 2.9 Software Taxonomy [44]

implies the freedom that the users and developers could use the code or program for [40]:

- executing the program for one own's purpose,
- study how the program operates and modify it to one own's needs,
- redistributing copies of the original or the modified, improvised program, and
- improving the program and publicize it for the community.

According to Open Source Definition (OSD), a document published by OSI, there are ten criteria to decide whether a license of software is open-source which is shown in Figure 2.10 [40] [43].

1. **Free redistribution:** the software to be available for distribution without payment.
2. **Source code:** the soft to be distributed with the source or well-publicized access to it.
3. **Derived works:** the license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
4. **Integrity of the author's source code:** distribution of "patch files" used to recreate derived works to be permitted.
5. **No discrimination against persons or groups:** the license must not discriminate against any person or group of persons.
6. **No discrimination against fields of endeavour:** for example, it may not restrict the program from being used in a business, or from being used for genetic research.
7. **Distribution of license:** the rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8. **License must not be specific to a product:** license rights must not depend on the software being distributed with the other specific software.
9. **License must not restrict other software:** the license must not place restrictions on other software that is distributed along with the licensed software.
10. **License must be technology-neutral:** No provision of the license may be predicated on any individual technology or style of interface.

Figure 2.10 Open Source Delivery [40]

There are three critical benefits among the numerous benefits that could be found along with the emergence of OSS [41]. The first benefit is the fact that it reduces the cost for an IT company exponentially, the cost of development, and the innovation cost [41]. As it has the contribution from proficient developers in the community throughout the world [41]. Second of all, OSS can assist organizations, specifically small-medium enterprises (SMEs) and public institutes, to implement IT systems into their business process [41]. SMEs and public institutes may not be able to implement the IT systems due to the high cost of implementation and their limited budget [41]. Last but not the least, the information and knowledge from getting involved with the OSS project [41]. It is valuable to those who wish to increase their proficiency in programing language as they can learn through practice [41].

An abundant amount of individuals has involved themselves with OSS projects for their purpose and goal [41]. There are two key motivations as to why they are willingly contributing to the OSS projects, which are regarded as intrinsic motivation and extrinsic motivation [41]. Intrinsic motivation is driven by an individual interest or enjoyment of the project or task instead of relying on external pressure or rewards at the end of the project or task [41]. The research found that 54% of the individuals, especially young adults find enjoyment and satisfaction that they have during the programming and involvement in OSS projects [41] [44]. Their creativity and ideas that are implemented in the OSS projects, conferred a great sense of accomplishment motivated them to contribute to it [41]. Contrary to intrinsic motivation, extrinsic motivation is driven by an outcome such as profits and reputation [41]. Strictly speaking, the motivation will disappear once those rewards are withdrawn [41]. Some large OSS projects and contributors have been analyzed that the contributions have been involved with the pursuit of rewards and reputations [41]. Aside from those, research also found that a sense of reciprocity and job opportunities are the other factors of extrinsic motivation [41]. Some individuals would aid the OSS community by responding to the question in the community for them to be able to obtain assistance from others when required in the future [41]. Moreover, some IT companies would like to recruit capable programmers from the OSS communities which encourages them to be involved in the OSS projects [41].

All software, including OSS, is subjected to copyright law [43]. The copyright holder is an individual or organization that has control over the work, they could give permission to other individuals to use, modify, and redistribute their software according to specific provisions through licensing [43]. OSS license is different than

the general commercial license as it grants more access to the individual, allowing access to the source code and the right to modify it [43]. Copyleft, enforced by copyright law, is a term established by the free software community for a license condition to ensure that all versions of modification made to the source code can be copied, modified, and redistributed similar to how it is with the original [43]. There are frequently used open-source licenses for instance the Berkeley System Distribution (BSD) license, the Massachusetts Institute of Technology (MIT) license, and Apache. The BSD license is one of the least restricted as well as most acknowledged open-source licenses [43]. It allows redistribution of source code, modified or not,s long as the work has its copyright notice regarding disclaimer and limitation on liability found in the license [43]. MIT license is quite identical with BSD license as it authorizes the reuse of open source code if the license includes proprietary software in their term [43]. Moreover, it also allows the use of the copyright holder's name for the promotion of the software [43]. Vuejs and vuetify are examples of an OSS project released under the MIT license [45]. Apache is similar to BSD and MIT licenses, as it grants the use of software without the obligation to redistribute the code of any version modification of the software [43]. Aside from that, it supports the clause of patent licensing and termination [43]. Understanding the difference between each licensing is essential in order to use OSS.

2.8 Software Development Life Cycle

The concept of SDLC could be used to develop a website application. Software Development Life Cycle which is commonly known as SDLC is a framework or a model that describes a continuous process of how the project started the deployment of the product [46]. Moreover, about planning on how to develop the product [46]. It

is a cycle that will help arrange the activities to be executed during the process of development [46]. SDLC approach to complete the development is by using a step-by-step approach, and the main phases are:

- specification,
- design,
- validation, and
- evolution [46].

The specification phases focus on understanding and comprehending the problem and what is required by the customer or the user in the application [46]. Designing is involved in planning out a solution for the problem described in the specification and designing how the application will be executed and flow [46]. The validation phase is in which the application is evaluated and tested whether or not it meets customer's expectations [46]. Finally, evolution is the latter stage of the project focusing on how it will be maintained [46]. According to the SDLC methodologies and purposes, these phases are described and utilized differently. Throughout the cycle, it will enhance the understanding of the product, the problem, along with the solution for the problem, programming, as well as how to test and maintain the product that is developed [46].

Working without thorough planning and discussion will complicate the process of developing an application with a team [46]. In addition, without a framework to structure the stages of development, developers will find it challenging to develop a product that they do not have a sufficient understanding of [46]. Thus, SDLC is an advantageous cycle to have in development [46]. Aside from that, SDLC aids in developing a timeline for the project, the responsibilities of each team member, as well as to keep track of the progress made during the development [46].

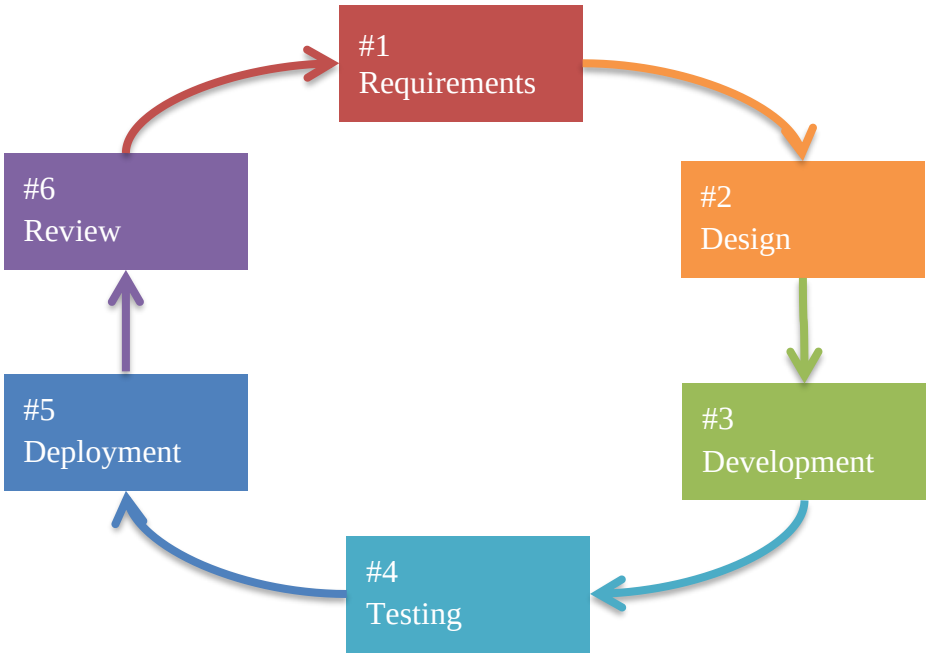


Figure 2.11 Software Development Life Cycle Agile Methodology

There are various SDLC methodologies. A traditional SDLC methodology is most known as the waterfall model and the modern models such as iterative model, agile model, and V-shaped model. Methodology like the agile model is excellent in order to reduce the amount of error later in the deployment stage [46]. Due to the fact that revision could be made over and over again during the implementation stage [46]. Agile methodology is a modern project management method in which the project is

made and adjusted based on customer or a user requirements for the project and allows quick revision and modification to be made instantaneously [47]. The agility in completing the project is acquired by including only the process in the project [46]. It entails the exclusion of all types of processes and activities that are irrelevant and will consume more time than necessary for the project [46]. Each iteration of processes involved in the agile SDLC is as shown in Figure 2.11, it consists of small and manageable which could be completed within a couple of weeks [46].

Each agile project regularly involves a customer or a stakeholder representative on the team [46]. At the end of each iteration, they will have to re-evaluate the progress and make some adjustments directly with the team if necessary [46] [48]. Thus, emphasizing the importance of enhancing their communication skill [46] [48]. Moreover, since more requests to change the requirement may be made by the customer, agile relies on the implementation of the software rather than the documentation [46]. This methodology offers more efficiency and faster project completion time [46].

There are 12 principles behind agile software development according to the agile manifesto [49], which are:

- satisfy customers throughout the delivery of the software,
- accept any changes in requirements throughout the development process,
- frequently deliver working software in a short timescale,
- customer or stakeholder representatives and developers should work together throughout the project,

- provide developers the environment, support, and trust to help them finish their job,
- relay information to and from the development through face-to-face conversation,
- the main assessment of progress is a working software,
- promote sustainable development,
- enhancing agility by continuous attention to outstanding excellent and design,
- simplicity is essential,
- good architecture, requirements, and design comes from good self-organizing teams, and
- adjust and reflect periodically.

The main advantage of agile methodology as briefly described previously would be that the team would be able to reduce the overall development and building time of the entire project [46]. In addition, customers and representatives would be able to get updated often after each iteration, making feedback, revision, and adjustment would be able to get implemented as soon as possible [46].

In agile methodology, there exists a Scrum framework to help increase the agility and flexibility in developing a project [50]. It is developed as an additional framework for the purpose of becoming a development methodology as well as a management methodology for software projects [50]. In Scrum, there are three roles, a product owner (PO), Scrum Master (SM), as well as the team [50]. A PO usually consists of a project manager (PM) and a business analyst (BA) who is responsible for defining and registering the requirements and the specification of the project [50]. The team consists

of developers, testers, and other relevant roles which will complete the backlog provided by the product owner [50]. Meanwhile, the Scrum master is the individual who will set the Scrum process in the project and make sure that everyone involved in the project implements the Scrum method provided [50]. The Scrum process is implemented as :

- determining the backlog,
- sprint planning,
- regular stand up meetings,
- sprint review, and
- sprint retrospective [50].

The PO will be the one to prepare the backlog for the team to follow, which the SM will arrange the priority of the task [50]. The whole team needs to participate in a meeting to evaluate the backlog and assign the task or feature that the developers will need to finish by the end of the sprint for the next sprint [50]. To supervise the progress, there should be a meeting to determine what each member of the team has accomplished prior to the meeting [50]. At the end of every sprint, the task will be demonstrated to the customer or the stakeholder representative in order to show the features or progress that have been completed [50]. Thereafter, the whole team would discuss the problem they encountered while doing their task that would need to be discontinued in the next sprint [50]. By using the Scrum framework, the quality and progress of the project could be seen clearly and done more efficiently [50].

2.9 Design

Design defines the architecture of the system in the early phase of SDLC. The system structure is designed based on the requirement analysis [51]. The logic, architecture,

physical design, and final design may be created in the design phase [51]. The design is used to sketch the overall picture of the system for the team [52]. To design the system, one should consider both the functional and non-functional requirements of the system [52]. Visualizing the design aid the team developing the system to picture the process of the system [52]. To visualize a system, diagrams is necessary [52].

2.9.1 Use Case Diagram

There are various diagrams that could be used to visualize a system. One of them is Unified Modelling Language (UML). UML is used as a visual language to interpret and document a system as well as to present how the user could use the system and the behaviour [53] [54]. There are several types of UML diagrams that developers could use such as class diagrams, activity diagrams, use case diagrams, and component diagrams [54]. The frequently used UML diagrams are:

- use case diagram,
- activity diagram,
- class diagram,
- sequence diagram,
- collaboration diagram,
- component diagram, and
- deployment diagram [53].

Each type of UML diagram has a different perspective for the customer and developer [53]. Use case diagram is one of the types of UML diagram used to aid the developers in illustrating the functionality of the system [53]. This includes the function requirements, the relationship between the ‘actors’ and the functions as well as the

expected process of the system [53]. The actor is the individual who is interacting with the system [53]. This diagram shows a certain group of use cases with similar functionality.

2.9.2 Flowchart

Meanwhile, the logical design of a system could be represented by a flowchart [55]. Contrary to the use case diagram, a flowchart is a diagram that portrays a process, system, or computer algorithm [56]. This diagram is commonly used in a variety of fields, from documents, study plans, to complex processes in a comprehensible diagram [56]. A flowchart uses several shapes to define its functionalities and processes, such as rectangles, diamonds, and ovals which are connected to one another [56]. Moreover, since flowchart can be used in a variety of forms, it is one of the most frequently used diagrams, used for both technical and non-technical fields [56]. The flowchart is used as a guide for developers to develop the system. [55].

2.9.3 Wireframe

A wireframe is convenient to discuss and improve an application's UI [57]. It presents the outline of an application interface and is used as a visual guide of the contents and structure of the frontend side [57] [58]. Wireframing is the first step in the UI design process, giving a guide for the layout and functionality of the client-side [59]. It allows the developers to be certain of what the final product will look like [59]. Wireframe provides a way to communicate for the client and the designer to be certain of what the client requested [59].

2.9.4 Sitemap

A sitemap is a map of every page or URL available on a website [60] [61]. By providing a sitemap to a search engine, it is easier for them to index the sites and return search results based on a keyword, helping the user to find the website [60]. Sitemap contains relevant information about a website, for instance, the landing page, how each page relates to the other, and the structure for search engines to make sense of the website [60] [61].

CHAPTER 3
PROBLEM ANALYSIS

This chapter will delve into the competition of the website application briefly explained in Chapter 1. In addition, the author will also discuss the problem analysis, how the website application is different from the existing solutions and the solution to the problem stated in Chapter 1. Moreover, this chapter will explain justifications for the tools that the author decided to use.

3.1 Problem Analysis

As introduced in Chapter 1, the PO of this project is Mr Ardimas, while the client is Mr Widodo Nugroho, the founder and owner of PT. GTN, introduced the problem to the author's team. As represented in Figure A. 4, the problem presented for the author's team is to make a cost-effective application for the client's company. The detail of the requirements of the application are also presented in Figure A. 4. In particular, the author's problem is to create the first and fourth features stated in the requirements.

The author's responsibility based on the requirements in Figure A. 4 is to fulfill the requirements of:

- visualization of oil and gas data,
- file management to store the user's file, and
- creating a UI to accommodate the prediction model to predict the oil and gas production.

The visualization of oil and gas data is the essential part of the application as users would like to focus on the use of this feature, without the visualization, the application will have no meaning. The file management is an additional feature so that the user would be able to store their file and use them in the application, additionally, this means that the user could access the files anywhere without exchanging files or data between devices. Creating a UI for the production model is also crucial so the user would have an easier access to the prediction model.

3.2 Existing Solutions

The existing solutions are researched based on the data visualization and the file management features which are within the author's scope and responsibility. The data visualization feature is to help comprehend the data better by converting the raw data into a chart visualization. While the file managements feature is for the user to group their files and use them in other features conveniently. These features are essential for the applications as it aimed to aid the user with visualization.

3.2.1 Delfi Virtual Data Room

Schlumberger provides the technology and services for the energy industry [62]. One of the services the services that they have is the Delfi Virtual Data room which allow user to upload documents and placing them in an online repository [62]. Moreover, the documents they stored in the online repository could be downloaded and used in other services available in Schlumberger for research and analysis on the document [62]. This service was created for experts to evaluate the document virtually which the document is shared in a secure environment [62].

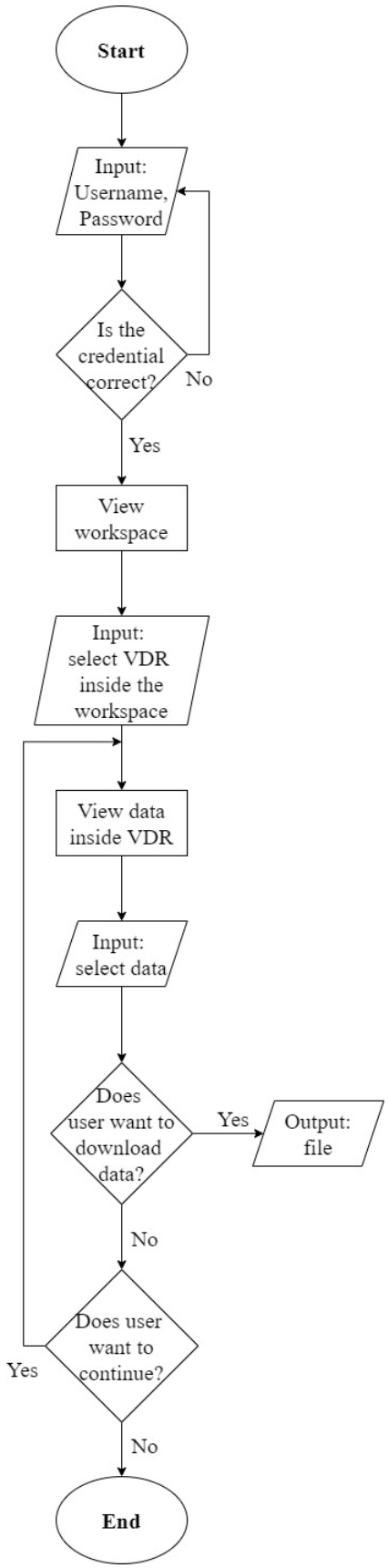


Figure 3.1 Delfi Virtual Data Room Process

According to the demo video [62] they show, the process is described in Figure 3.1. After the process is done, the file downloaded can be used in Schlumberger other services, which is Petrel and Techlog [62].

The advantages of using this application are:

- the data rooms allow the user's assets to global audience and
- provides access to the petrotechnical suits in the Delfi environment [62].

However, although the application allows the data in the workstation to be fully configured to the petrotehcnical suits they provides, the Petrel and Techlog, the user is required to download the data first from the data room before using it anywhere else.

3.3.2 Ideals Virtual Data Room

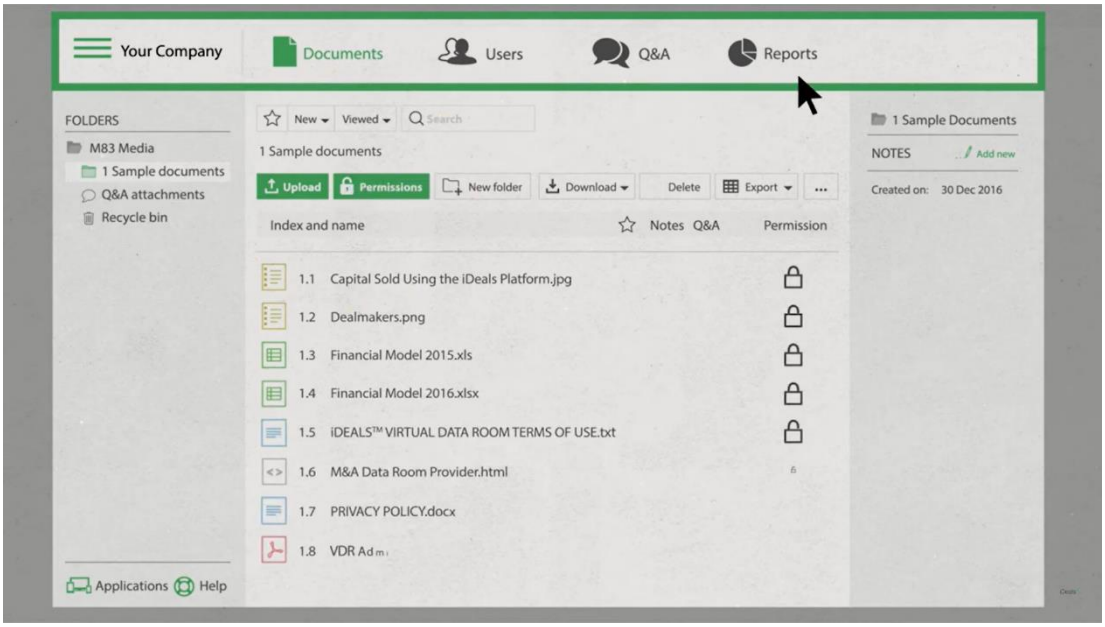


Figure 3.2 Ideal Virtual Data Room [63]

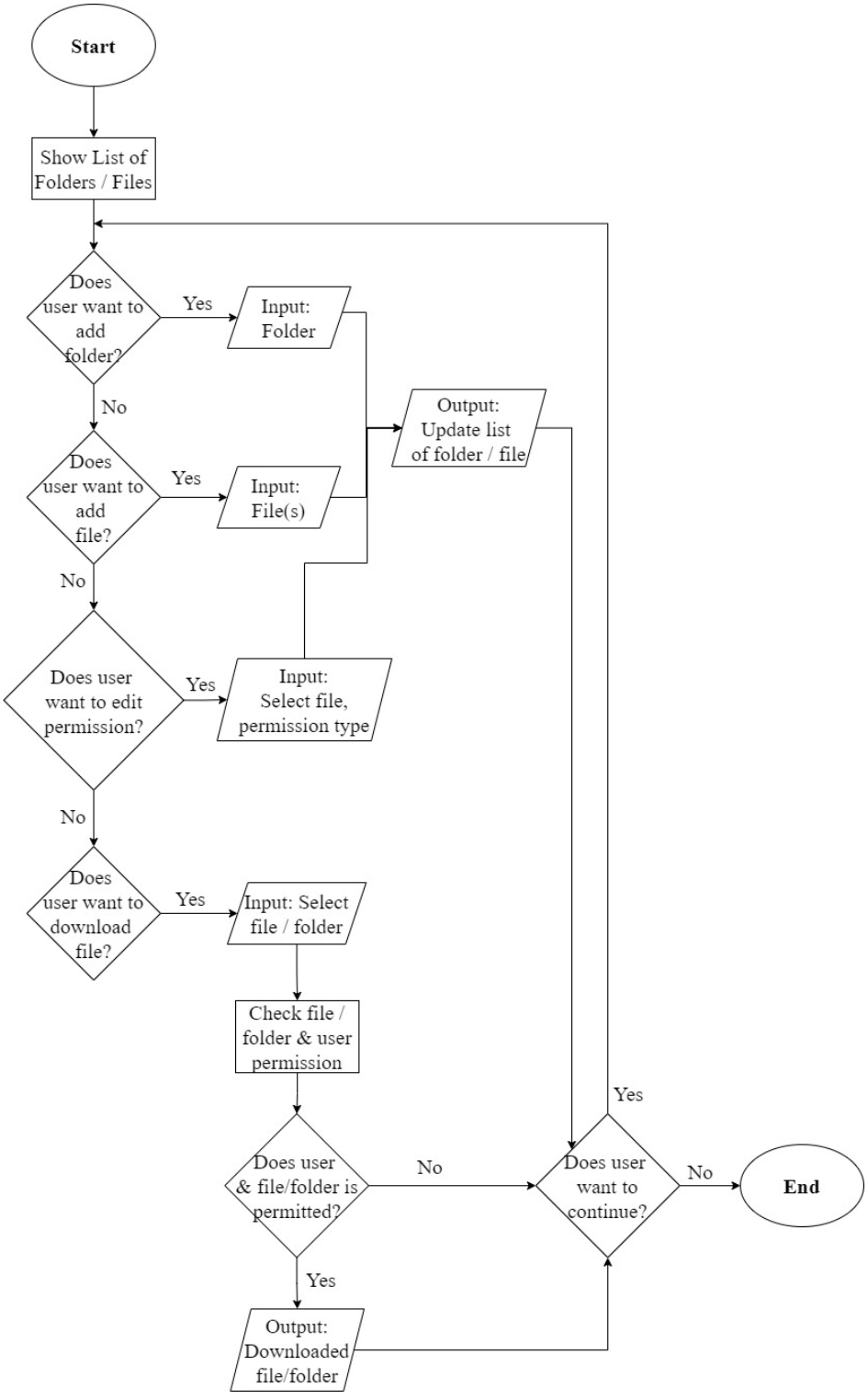


Figure 3.3 Idelas Virtual Data Room Process

Ideals VDR focus on the sharing of documents in a secure way and not for the energy industry [63]. Mainly focusing on the sellers and buyers, providing space for sensitive documents and deliver them to hundreds of buyers simultaneously in a short period of time [63]. The VDR is as shown in Figure 3.2.

Based on an introduction video [64] they provide, this application works as described in Figure 3.3. The advantages of this application are:

- setting up document permission and
- allow access on any devices [63].

However, the disadvantage of this application which is not build for the energy industry is not being able to integrate the data fully to petrotechnical solution.

3.3 Proposed Solutions

To elaborate more on the solutions proposed in Section 1.2.1, the solution is to develop a VDR website application with a lower cost. To lower the cost of development, OSS will be used. As mentioned in Section 2.8, OSS could greatly help SMEs who could not afford to build an application for themselves. Furthermore, developing a website application means that the requirement for an OS is not liable for consideration as a web browser could exist in most OS, such as, Windows, Linux, MacOS, Android, and iOS. Users would be able to access the application through their mobile devices as well

With the OSS, the author's team can develop the visualization of oil and gas data, using 2D or 3D visualization. Moreover, it will help in developing the file management

system and the UIs for the website. The visualization is going to help the user comprehend and analyze the oil and gas data that they have.

The VDR website application also allows future customization. Some oil and gas companies request to have their own customized VDR website application, for instance, an integration with a showcase-like application, or handpicked features from the existing application. Hence, the author's team decided to develop this website application to help the oil and gas industry in Indonesia.

The VDR website application can be compared to the existing solutions explained in Section 3.2, although there are similarities, how the features are used are different. The common features between the VDRs are the file management system, allowing user to access the data. The differences between the applications are describe in Table 3.1

Table 3.1 Table of Comparison

Features	Delfi VDR	Ideals VDR	VDR web app
File Management	✓	✓	✓
Integration with petrotechnical solution	Data are fully integrated but used in another software	No integration at all	Data and petrotechincal solution on the same application
Security	Username and password	Single sign on	Username and password (OAuth2)

Permission	Allow user to access workspace	Assign permission type to user, file/folder	Assign user permission type

CHAPTER 4
SOLUTION DESIGN

This chapter describes and presents the visualization of how the website application works. How it interacts with the user and the server with the use case diagram, the flow of the interaction of the user with the website application with the flowchart, the UI design, as well as how the test phase are going to be conducted in this thesis. It focuses on visualizing the design of the website application at the initial phase of the SDLC, which is the design phase.

4.1 System Architecture

The system architecture of the website application is depicted in Figure 4.1. The users would be able to access the 6 main components which are the features of the website application. The users in which it could be a regular user, premium user, or administrator. The author is responsible in developing the frontend for four of the features as shown in Figure 4.1, which are the:

- administrator page,
- viewer page,
- file management page, and
- production page.

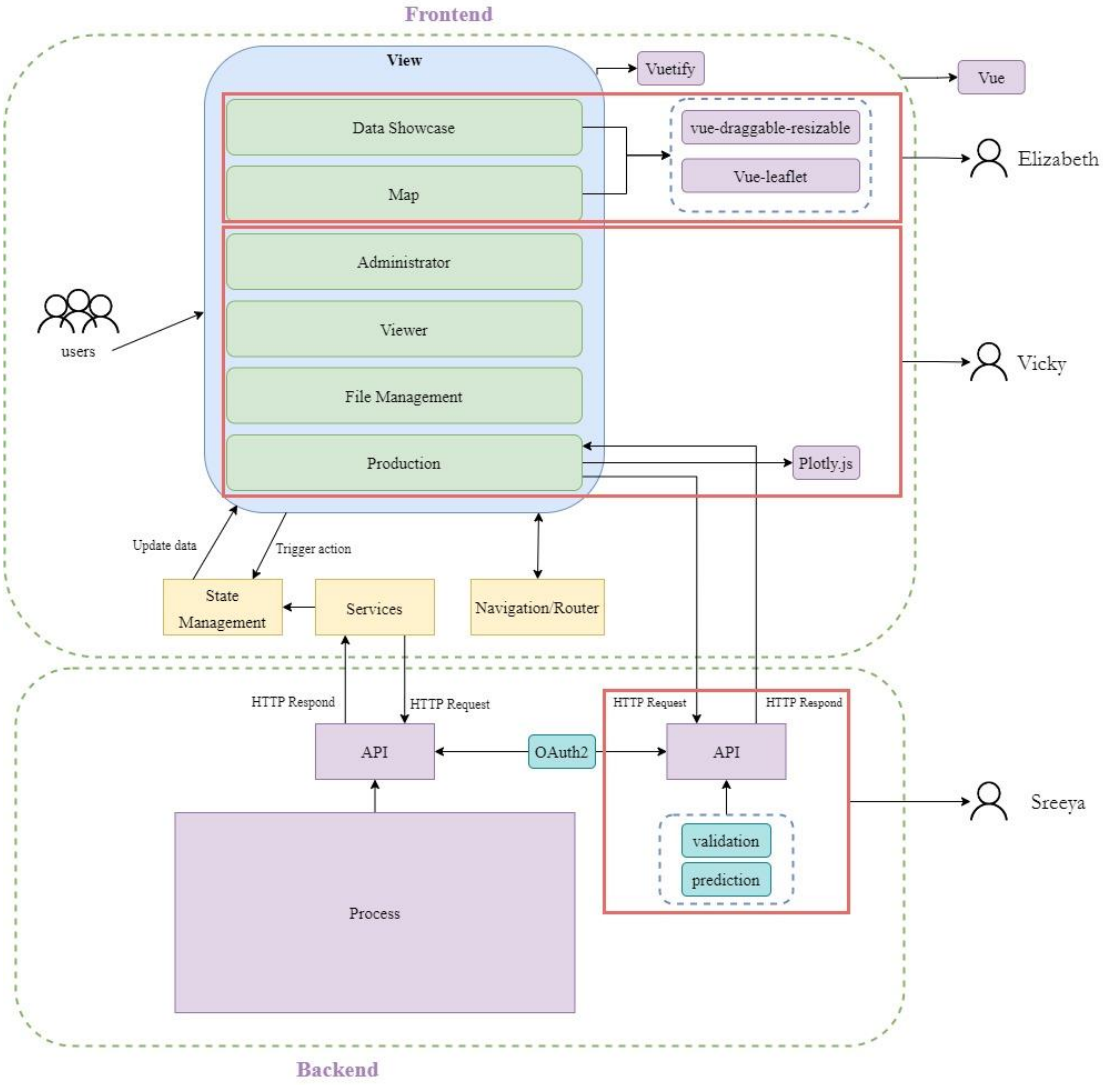


Figure 4.1 System Architecture

4.1.1 Technologies

There are a lot of open-source libraries and frameworks available to build the features that this project focused on. The website application will be made a SPA with VueJs as the framework. VueJs is an open-source JavaScript framework licensed by MIT. As for the UI libraries for the frontend development, the author decided to use Material Design by Google. Vuetify is a Vue UI library for Material Design framework released under an MIT license. The latter subsections will further delve into the reason why the author choose the frameworks and libraries for the project.

For the production page, the author is using an open-source library Plotly.js to visualize the data. The ‘Services’ will be handling all the request and response to and from the APIs, and when necessary, the response will be directly directed to the state management.

4.1.1.1 Frontend Framework

Although there are other SPA frameworks such as Angular and React, the author decided to use VueJs instead. Aside from the advantages of Vue mentioned in Section 2.2.1, it is the most user-friendly and easier to understand for beginners who have just started to learn about frontend web development as depicted in Figure 3.3 [64] [65]. There are other additional advantages to using VueJs [64], such as:

- great CLI tool,
- Vuex for state management and Vue Router from the official Vue library,
- lightweight, and
- easy to understand documentation.

Meanwhile, React and Angular have their own disadvantages. React needs to import libraries for state and model since it does not implement MVC architecture [19]. Moreover, Angular has various structures compared to Vue, which means it is tougher to learn for beginners [19]. It is also hard to adapt to Angular as there are always significant updates [19]. As not every member in the author’s team had learned any frontend framework prior to this project, thus, the author decided to use Vue for being beginner-friendly [65].

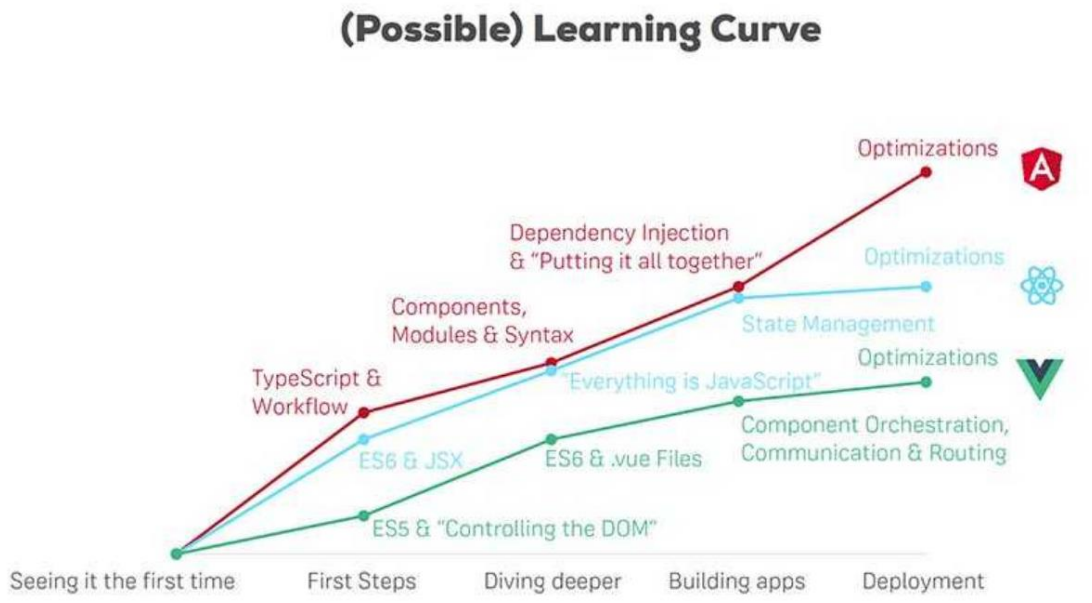


Figure 4.2 Frontend Framework learning Curve [65]

Moreover, for the navigation router, the author is using vue-router and vuex for the state management which are an official Vue library.

4.1.1.2 User Interface Libraries

Vue has a lot of in-built UI libraries, but the author decided to use Vuetify. As stated in Section 2.2.4, Vuetify uses the concept of Material Design from Google. It has advantages over the other Vue UI libraries, for instance, BootstrapVue, Buefy, Element UI, and Quasar [27]. These advantages can be depicted clearly in Figure 3.4. Compared to other UI libraries, it can be seen that Vuetify supports more features and supports. It is also easier as Vuetify has a large community to support the frontend developers. Moreover, rather than reinventing a component that already exist by using Vuetify, like pre-styled buttons, tables, and treeview, the author could focus more on the functions and features of the website application.

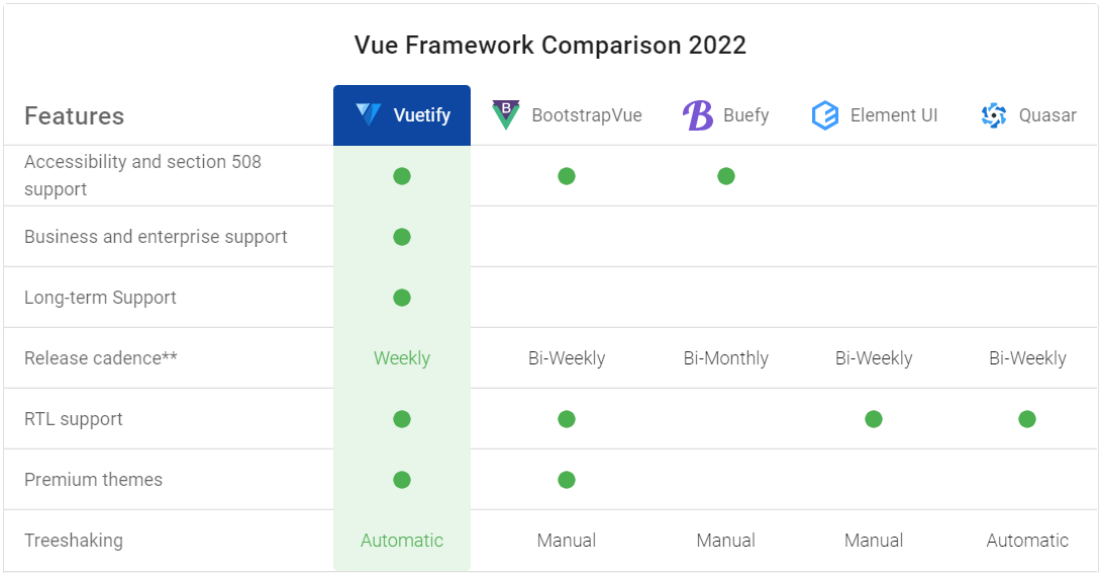


Figure 4.3 Vue User Interface Libraries Comparison [27]

4.2 Sitemap

Sitemap for users, premium user and regular user, is illustrated in Figure 4.4. Based on the sitemap, the author is responsible for developing the:

- viewer's page,
- production page,
- not-authorized page,
- file management page,
- login page, and
- profile page.

Sitemap for administrator can be seen in Figure 4.5, it shows the page that the administrator could access. The author is responsible to develop every page in Figure 4.5.

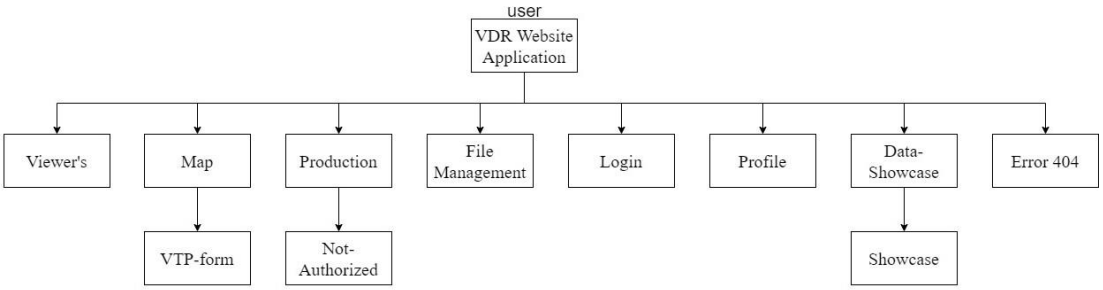


Figure 4.4 Sitemap for User

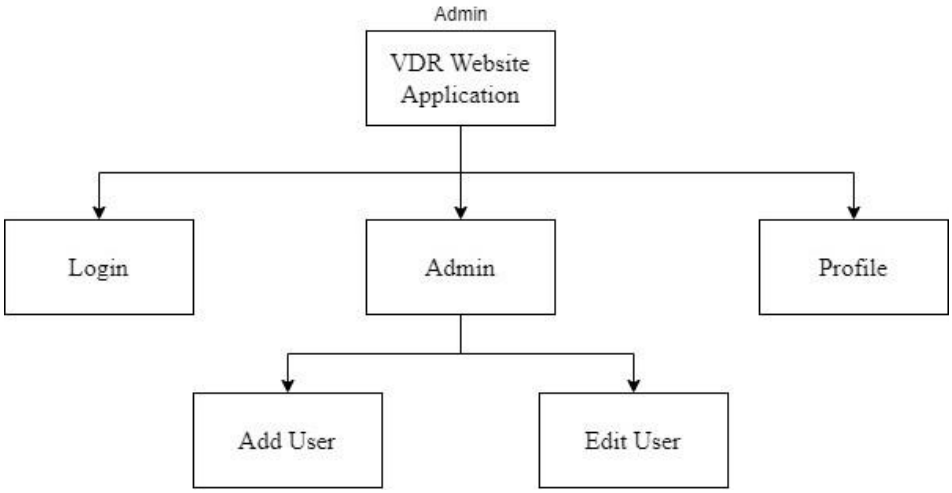


Figure 4.5 Sitemap for Administrator

4.3 Use Case Diagram

Figure 4.6 represents the UML use case diagram, the main actors in this Figure are the premium client and the regular client, which falls under the same category of a client/user. Everything inside the rectangle container represents the functions or the process that the client/user can perform. In Figure 4.6, there are a total of 10 processes that the client/user can experience in the VDR website application.

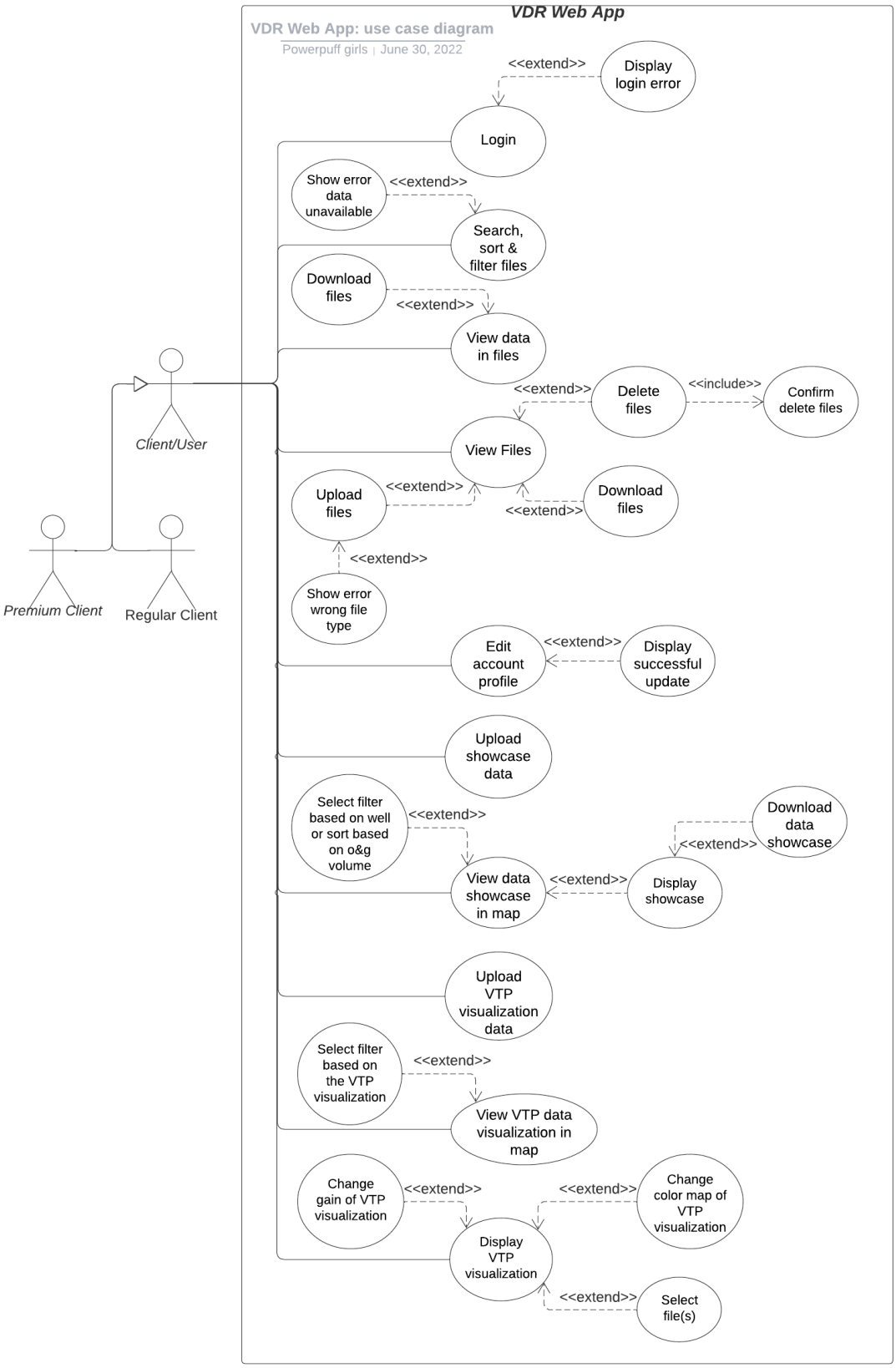


Figure 4.6 Use Case Diagram of a Client/User

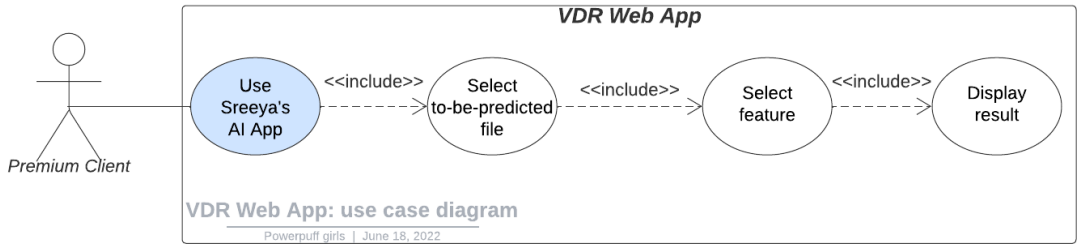


Figure 4.7 Use Case Diagram of a Premium User

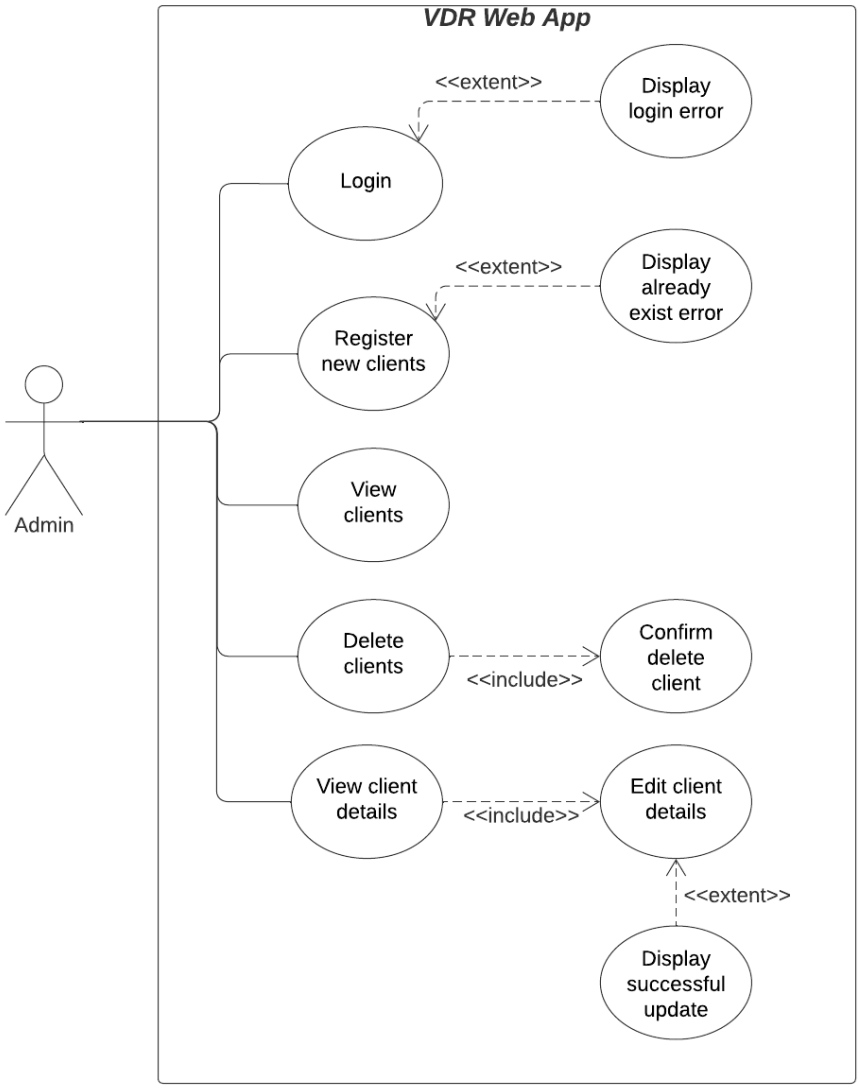


Figure 4.8 Use Case Diagram of an Admin

While Figure 4.6 shows every process that a client could access, Figure A. 2 present the process that is exclusive for a premium client. The process that is available to the premium client is accessing the predictive model that the author's team decided to develop and integrate into the VDR website application. Moreover, the process of the administrator in the website application can be seen in Figure 4.8. As administrators, they could access six processes which are all related to managing and modification of the client/user. As shown in Figure 4.6, there is no option for the user to register, a client/user would need to contact the administrator to get their accounts to be granted access to the website application.

4.4 Flowchart

Figure 4.4 presents the flowchart of a user, the process that the user could explore in the features that the author created. A full flowchart of the website application for a user can be seen in Figure A. 3. Inside Figure 4.4, there are rectangles with a vertical line on the left and right sides of the rectangle. This shape signifies predefined processors which are presented in Figure A. 1. Meanwhile, Figure 4.10 shows the administrator's flowchart, with their predefined processor defined in Figure A. 2 Predefined Processors part Figure A. 2.

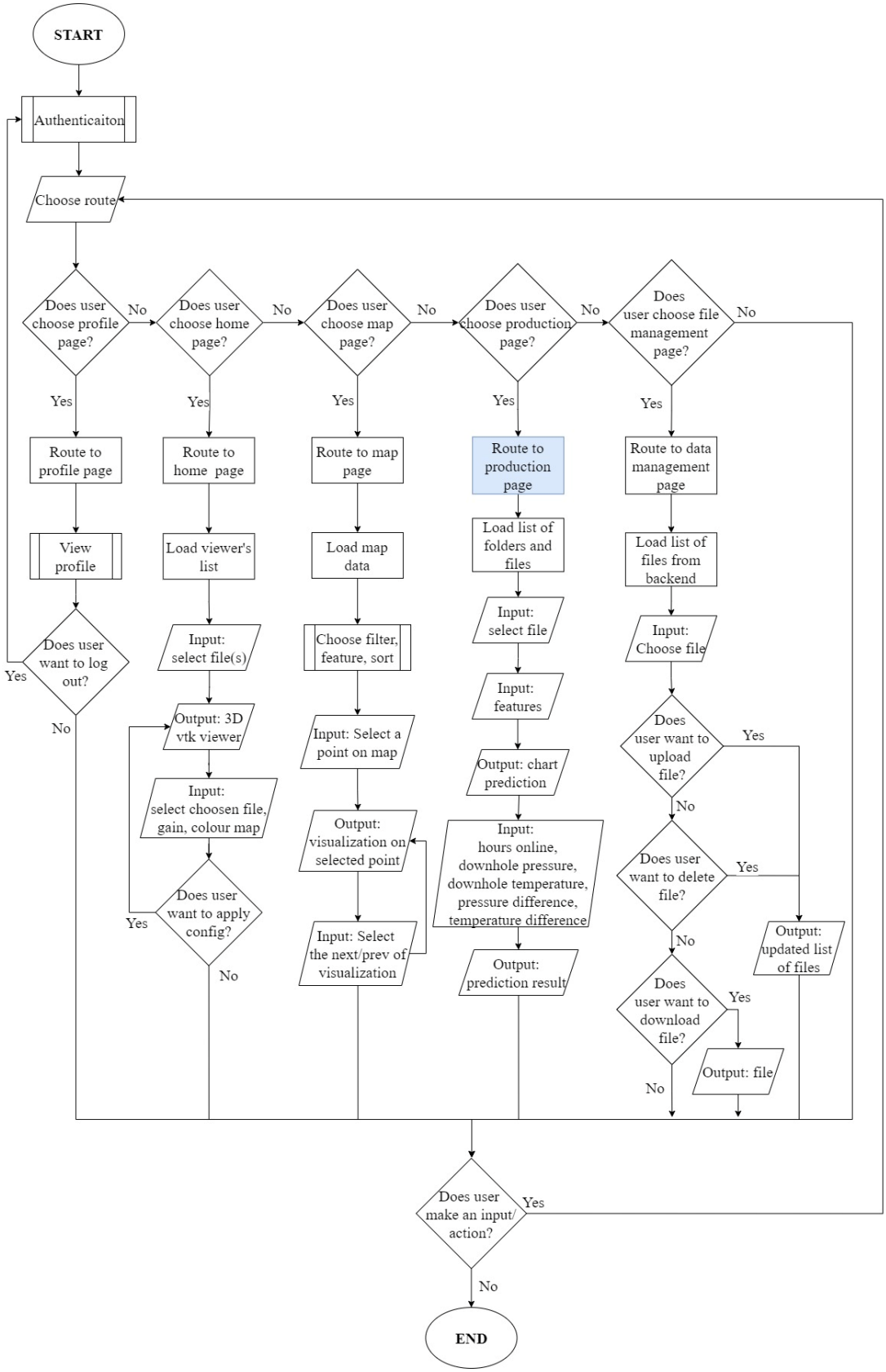


Figure 4.9 User Flowchart

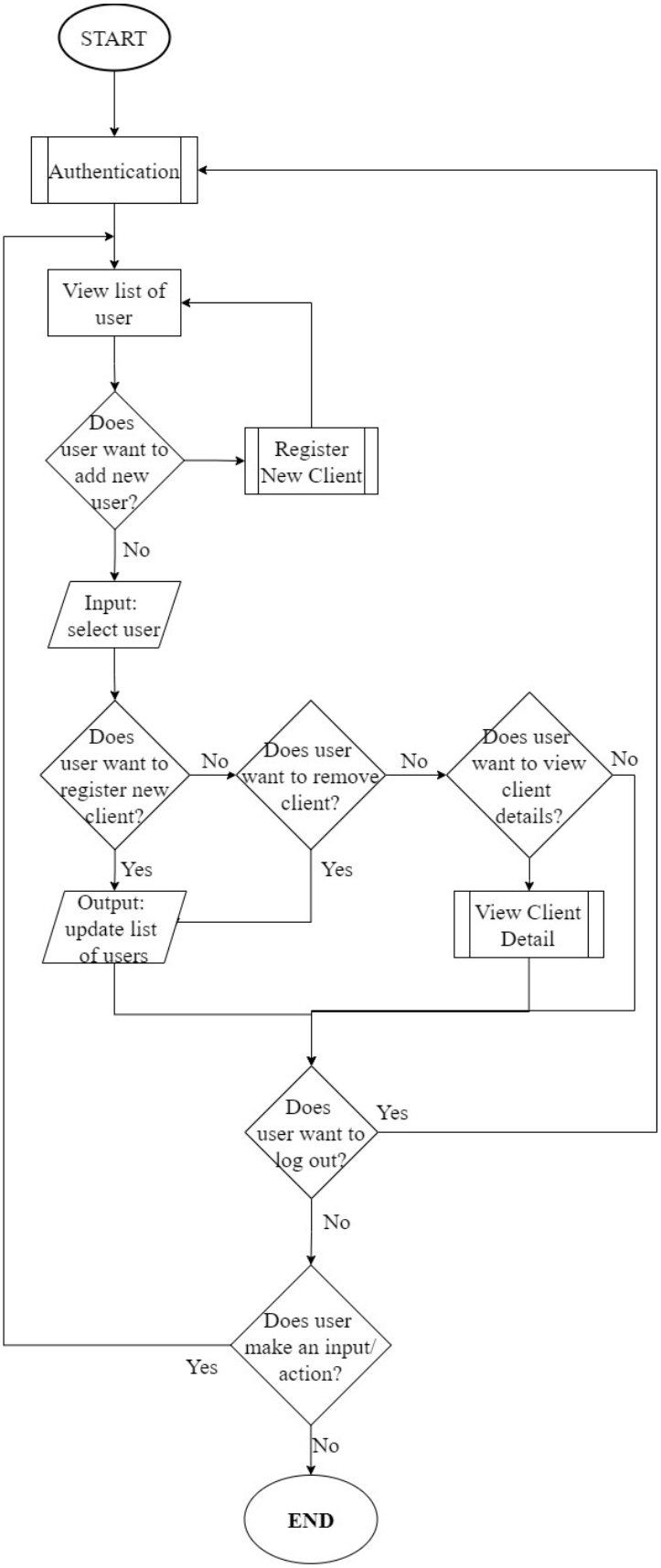


Figure 4.10 Administrator Flowchart

As depicted in Figure A. 3, after logging in, the user would be able to explore 5 routes, which are the profile page, viewer page, map page, map showcase page, production page, and file management page. Users would be able to view their account details on the profile page. The viewer page would be where the user would be able to visualize their .vtp file data. The map page is where the author incorporates the map developed by one of the team members, while the production page is where the author incorporates the prediction model. The user would be able to upload their files to the file management page. Meanwhile, Figure 4.5 represents what the administrator could do, as shown in the use case diagram in Figure 4.8.

4.5 Wireframe

The followings are the initial design of the website application.

4.5.1 Login Page

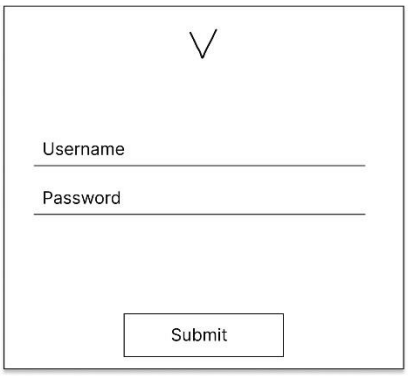


Figure 4.11 Login Page Design

The login page is as illustrated in Figure 4.11. After the user logged in, they will be redirected to the Viewer's page. Meanwhile, if the user typed the wrong credentials, a message will appear as depicted in Figure 4.12.

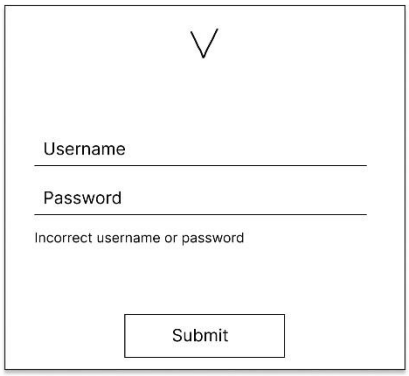


Figure 4.12 Login Page Error Design

4.5.2 Viewer Page

For the design of the viewer's page is as depicted in Figure 4.13, the main component of this page will be a card on the left side which will be consisting of the vuetify treeview to display the files and folders and the widget below it to modify the VTP. There are three things on the widget which are:

1. Select File

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

2. Gain

The gain is the data which will modify the texture of the data inside of the VTP file.

3. Colour map

   The colour map will change the colour of the VTP data.

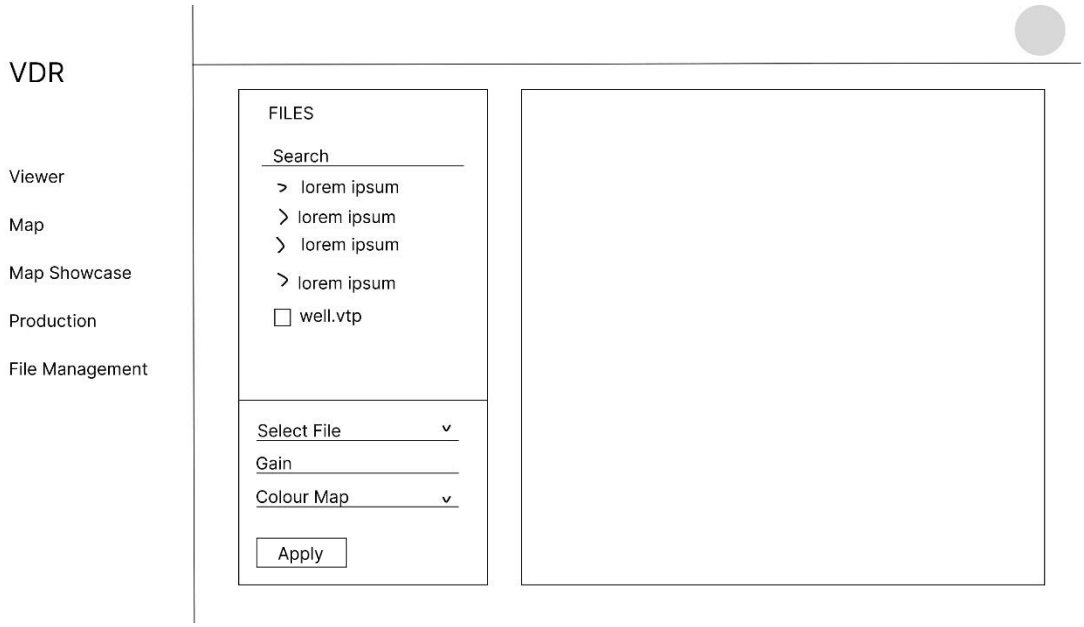


Figure 4.13 Home | Landing Page Design

When a VTP file from the treeview is selected, the empty card on the right side will be loaded with the visualization from the VTP file as can be seen in Figure 4.14. The VTP file will consist of the well log, seismic data, or surface for the visualization.

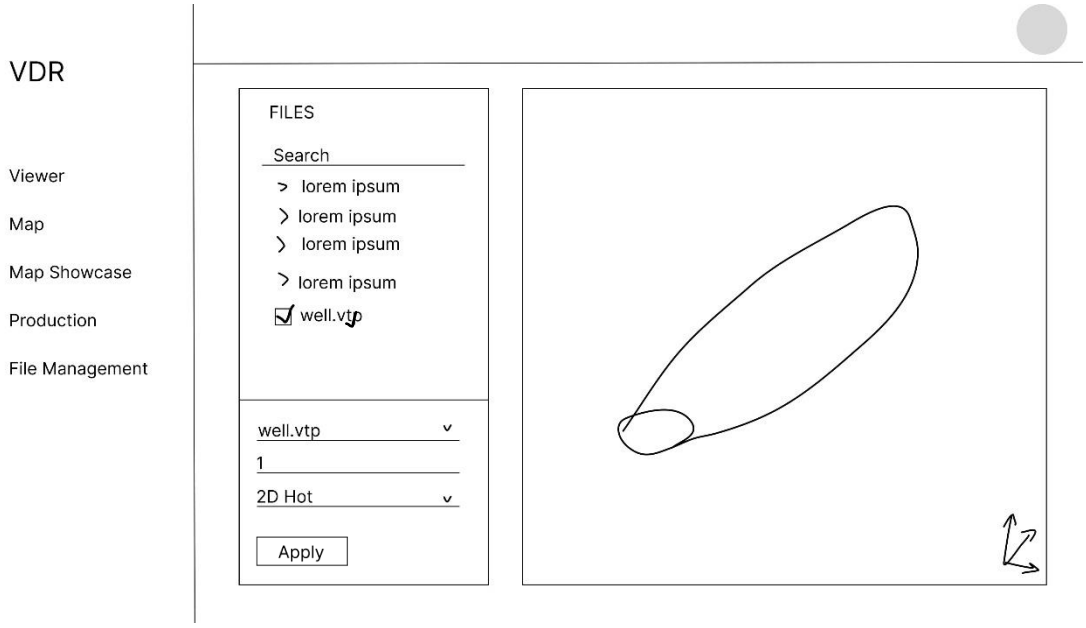


Figure 4.14 Viewer Select VTP File Design

4.5.3 Production Page

For premium users, they will have access to the Production's page. As illustrated in Figure 4.15, there are three steps. In the first step, the user will first have to choose the file that is in accordance with the template provided which is as depicted in Figure 4.16. The template is available for user to download in the file management. The files displayed would be based on the files the user uploaded to the file management with 'production' pointer, the pointer would be elaborated in a latter part. The file would only be able to be chosen if the file extension is .xls or .xlsx. Figure 4.17 depicted how the page would look like if there is no file presented with the 'production' pointer, the 'Add File' button will redirect the user to the file management page.

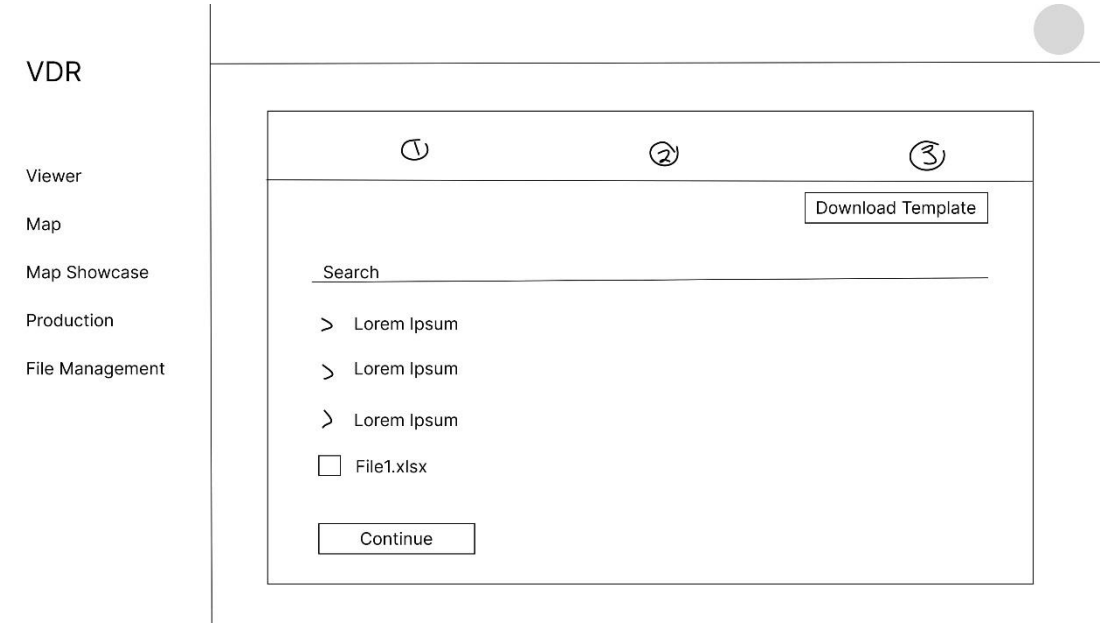


Figure 4.15 Production Page Step 1 Choosing File Design

	A	B	C	D	E
1	Hours Online / hours	Average Downhole Temperature / Deg C	Average Downhole Pressure / bar	Temperature Difference of the Well / Deg C	Pressure Difference of the Well / bar
2					
3					
4					

Figure 4.16 Production Page File Template

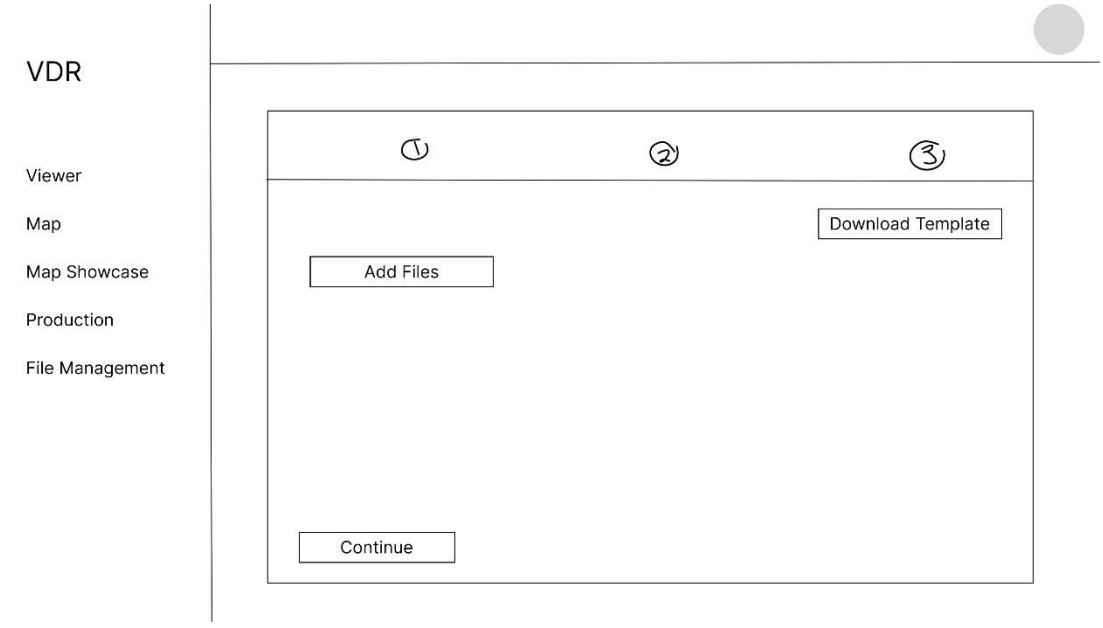


Figure 4.17 Production Page Step 1 No File Design

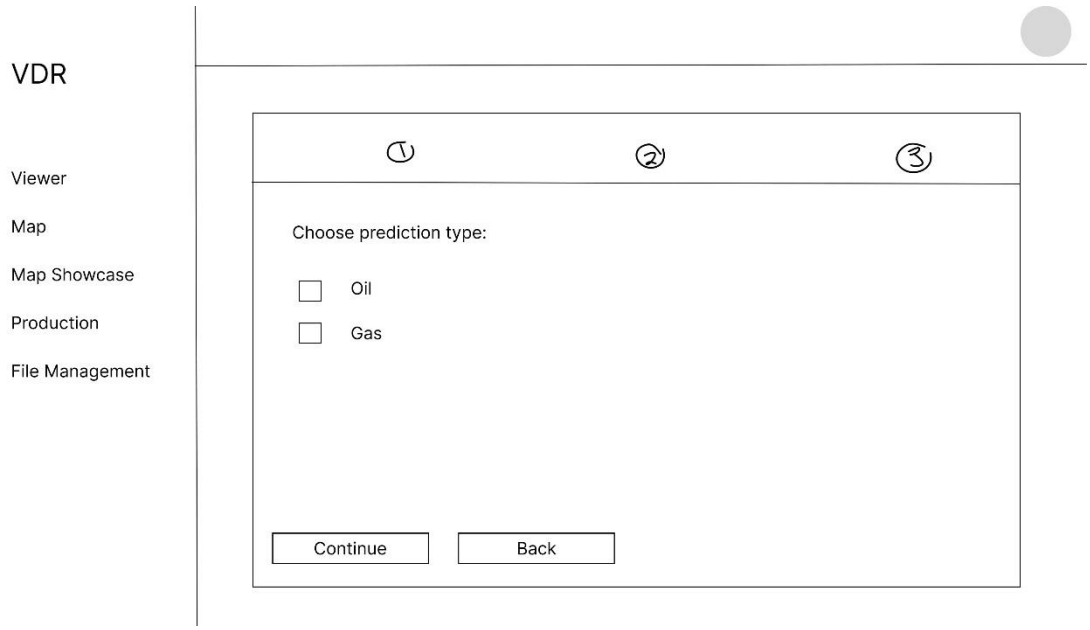


Figure 4.18 Production Page Step 2 Choose Feature Design

After the file is chosen and the user click continue, the user will proceed to step 2. In step 2, the user has to choose the features that they want to predict. As shown in Figure 4.18, the users can choose between two features, to predict the oil production or the gas production. The user could choose to go back and reselect the file they have chosen or click continue to proceed with step 3.

In step 3, the user will then be able to see the result in the form of a chart as illustrated in Figure 4.19. There will be three charts displayed, two 3D charts and a 2D chart to illustrate the prediction result along with the five features in Figure 4.16. Moreover, the user would be able to make a single prediction in case they want to make an elaborate search on the data. The 3D and 2D charts are displayed using Plotly, an open-source library under the MIT license.

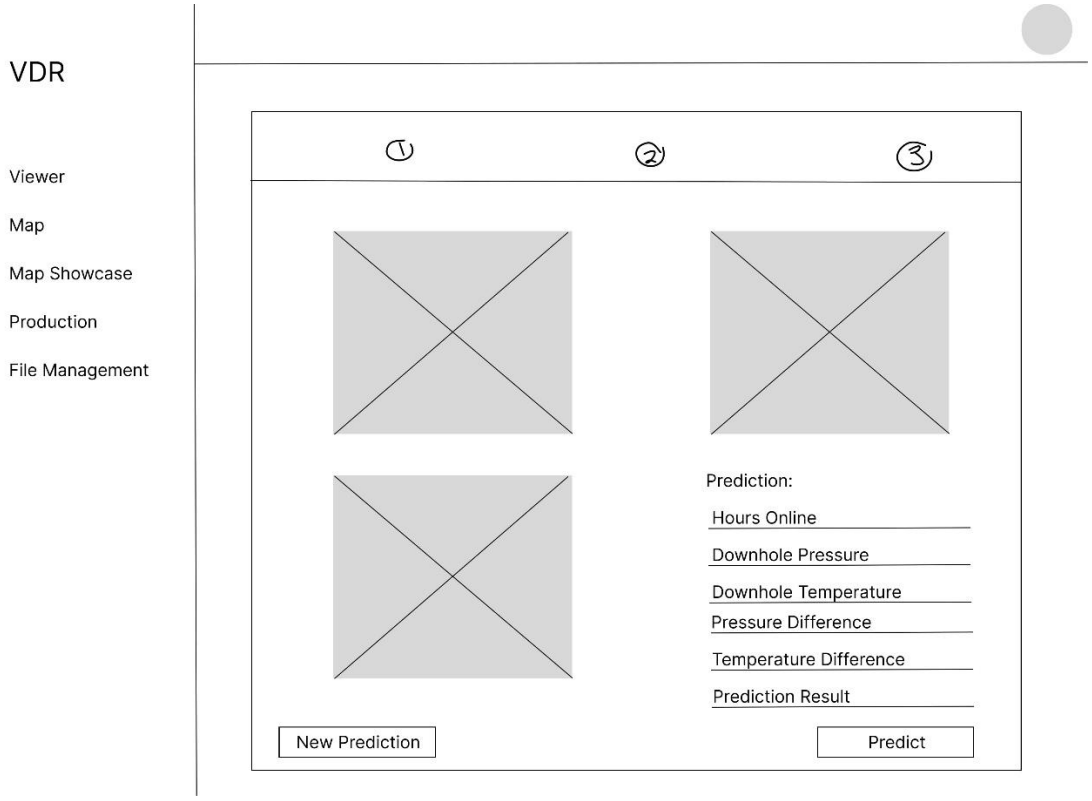


Figure 4.19 Production Page Step 3 Result Design

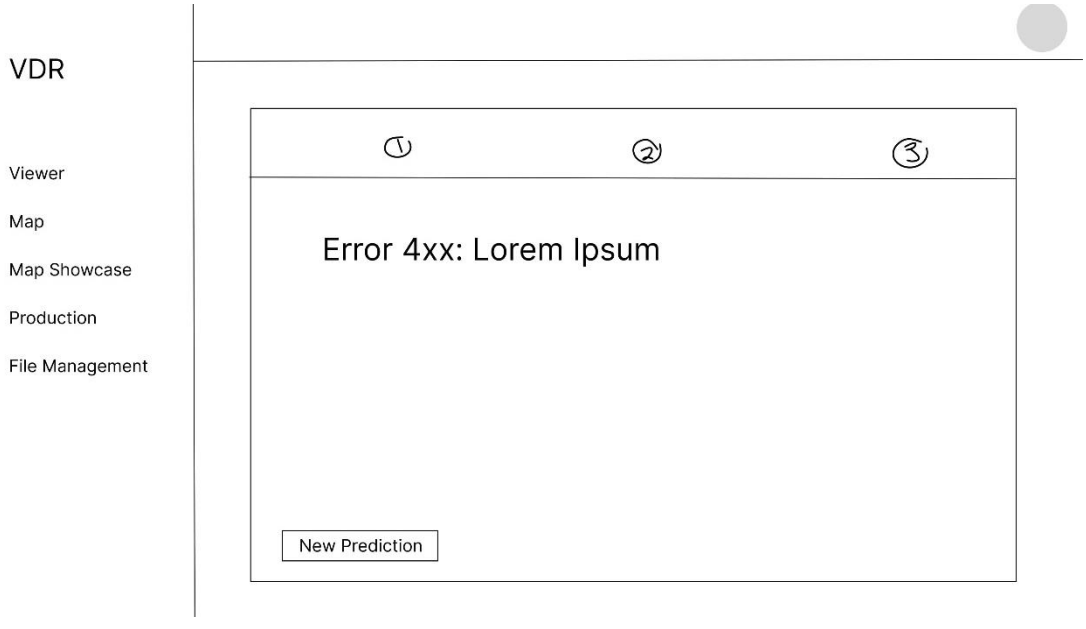


Figure 4.20 Production Page Step 3 File Error Design

Supposed that the user chooses a file which is not in accordance with the template in Figure 4.16, an error will be displayed instead of the result as illustrated in Figure 4.20.

In the case that the user is a regular user, they are not authorized to the production page, they will be redirected to another page as described in Figure 4.21.

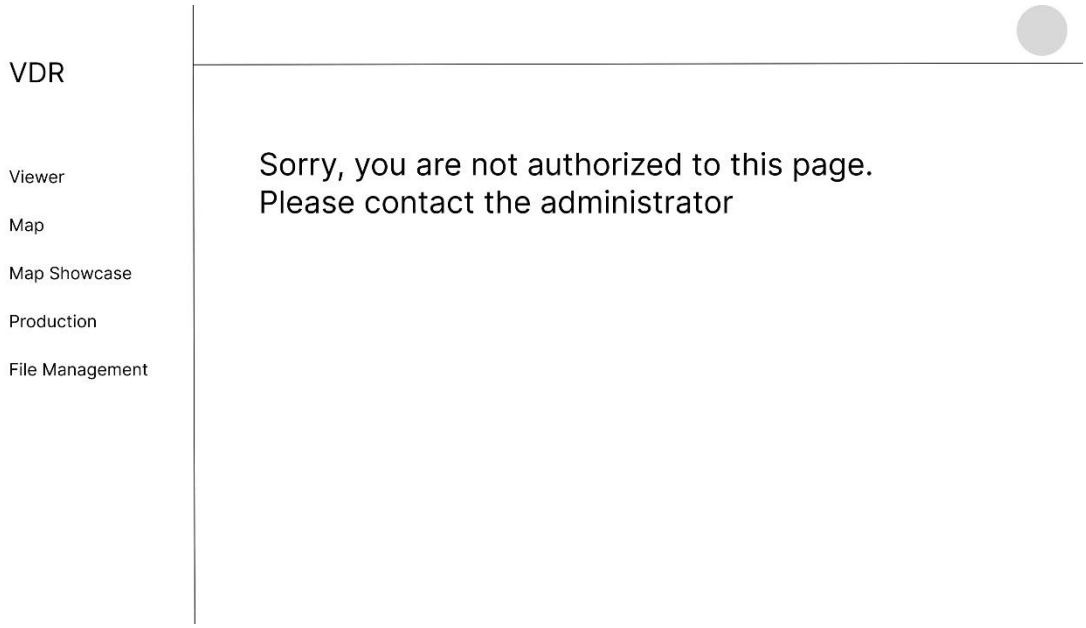


Figure 4.21 Production Page Not Authorized Design

4.5.4 File Management Page

As illustrated in Figure 4.22, the file management will be using treeview from vuetify, which make the user be able to put their files into folder. The folders have a dropdown button in between the checkbox and the text, which can be expanded if there is a folder or a file inside the said folder. If the row is a folder, the most right-side icon would be a delete icon, whereas if the row is a file, the icon on the right would be a download item.

If the user wants to delete a folder, a pop-up message as depicted in Figure 4.23 would appear. If the user click confirm, the folder, along with the files and folders inside it would be deleted. Clicking cancel would result to no changes.

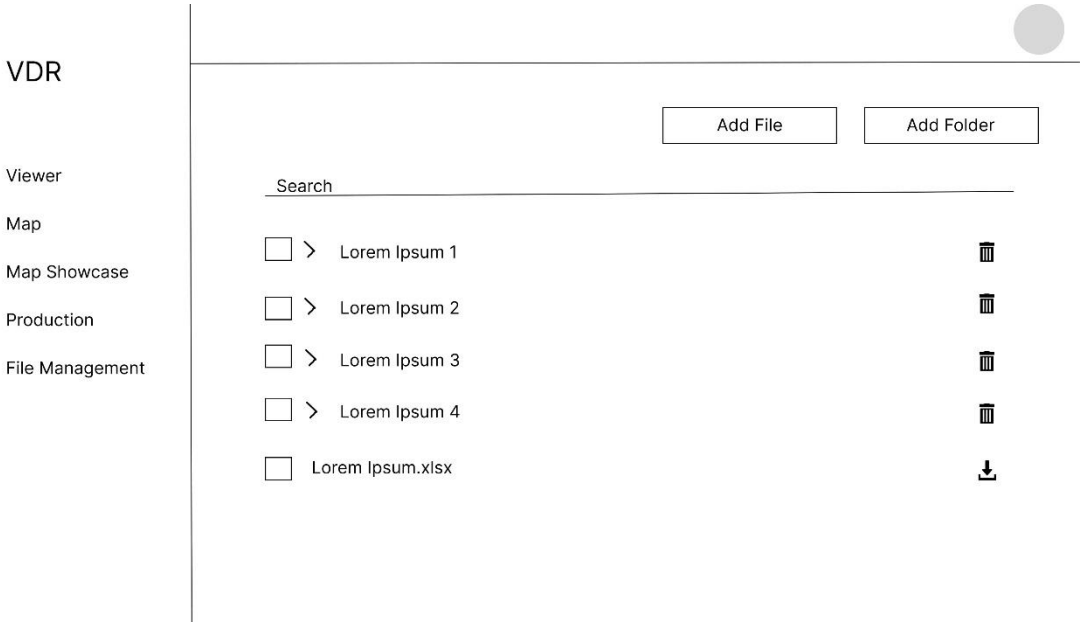


Figure 4.22 File Management Page Design

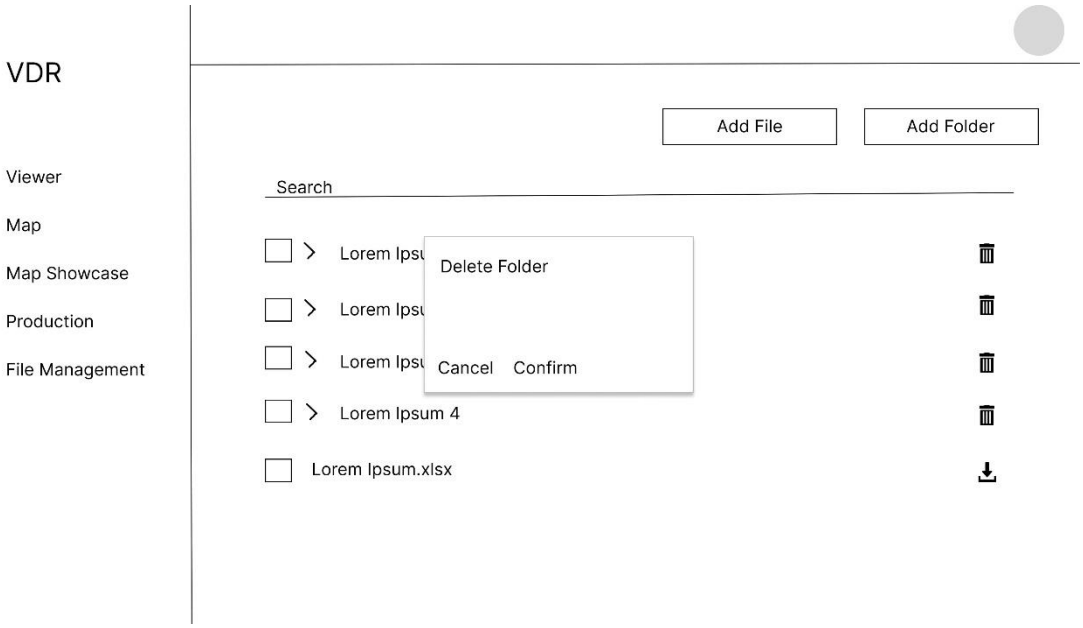


Figure 4.23 File Management Delete Folder Design

To add a new file, the user will have to press the “Add File” button, which the user will see a pop-up dialog to add their files as illustrated in Figure 4.24. By pressing upload file, the user will choose the files they want to upload. After choosing the files, the user must choose the category the files they chose belongs to. These categories will be a pointer for the files, there are several pointers which are:

- ‘*’ for general use,
- ‘showcase’ to be used in data showcase, and
- ‘production’ for production page.

After clicking save, there would be a loading animation showing while the files are sent to the API. After successful upload, the pop up will disappear and the files in the treeview would be updated.

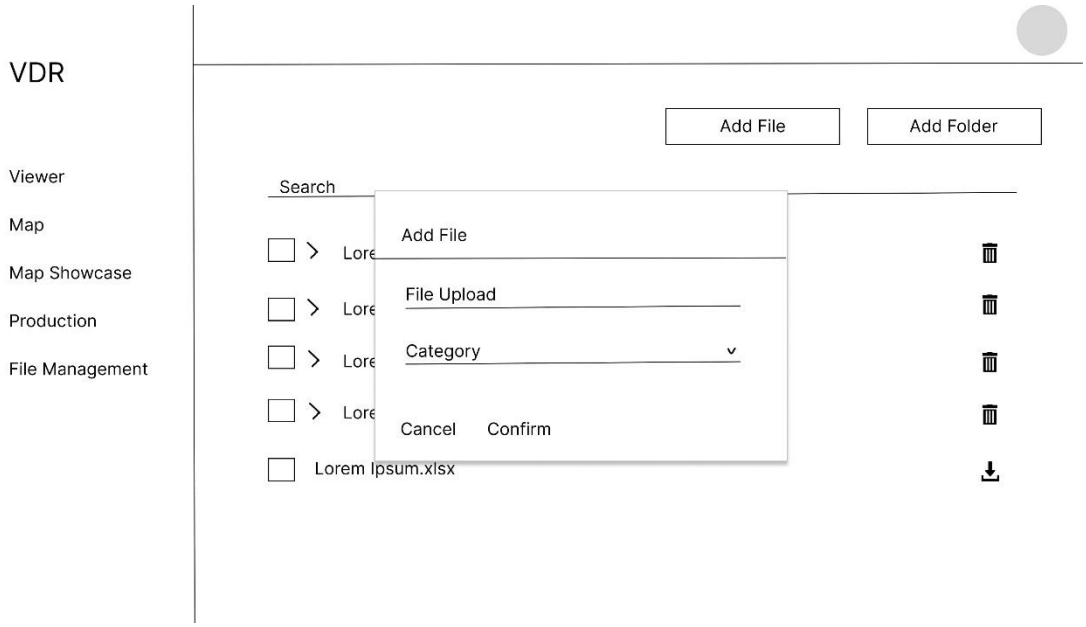


Figure 4.24 File Management Add File Design

To add a new folder, the user will have to press the “Add Folder” button, which will have another pop-up dialog as shown in Figure 4.25. The user only has to type the name of the folder and after adding it, a new folder will appear in the file management.

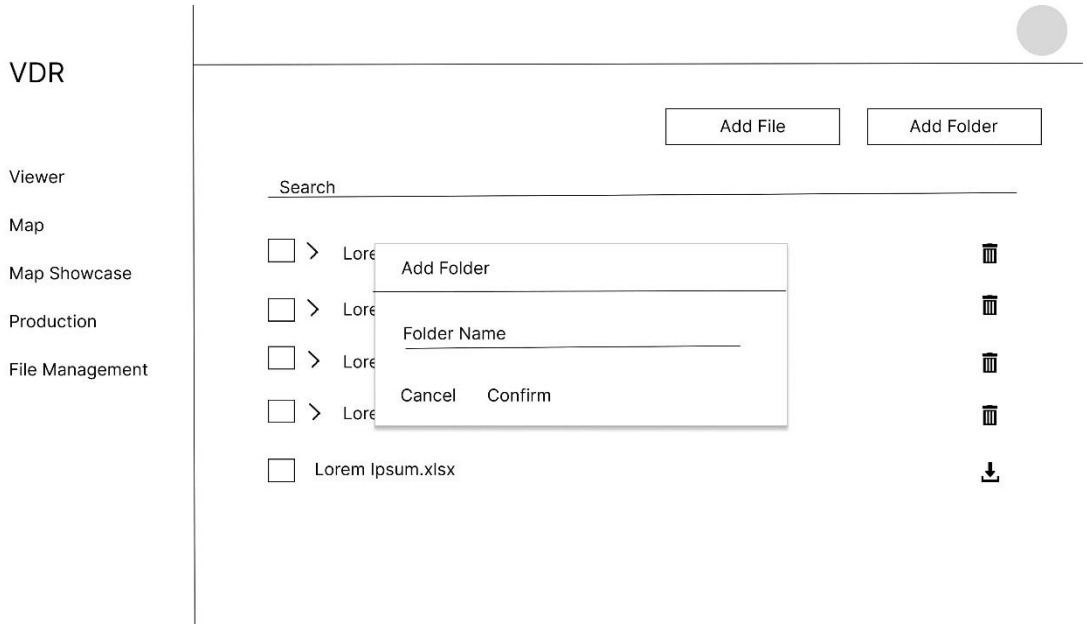


Figure 4.25 File Management Add Folder Design

To add a file or folder into a folder, the folder and file would be able to ‘activate’ as their row is clicked and change colour as can be seen in Figure 4.26. After the folder is activated, and the user press ‘Add File’ or ‘Add Folder’ button, the files they upload or the folder they add will be inserted into the folder they activated. The user does not have to activate anything if they want to upload to the root folder.

Figure 4.26 Treeview from Vuetify

To delete a file or multiple files at once would need the user to click on the checkboxes as illustrated in Figure 4.27. When pressing the checkbox of a folder, the items inside the folder would automatically be checked as well. However, although the folder is also checked, the items deleted would only be the files inside them. After pressing one of the checkboxes, a ‘Download’ and ‘Delete’ button will appear. Pressing ‘Delete’ button, a pop-up confirming if the user wants to proceed with the deletion will appear as shown in Figure 4.28. If the user press ‘Download’ all the files checked would be downloaded into a ZIP file.

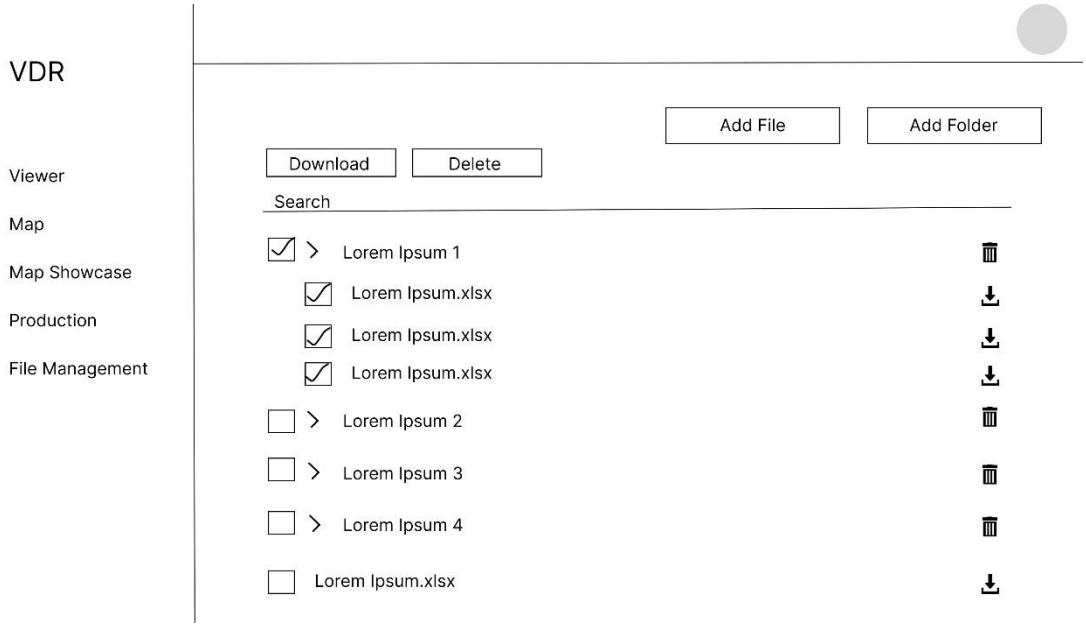


Figure 4.27 File Management Checkboxes Design

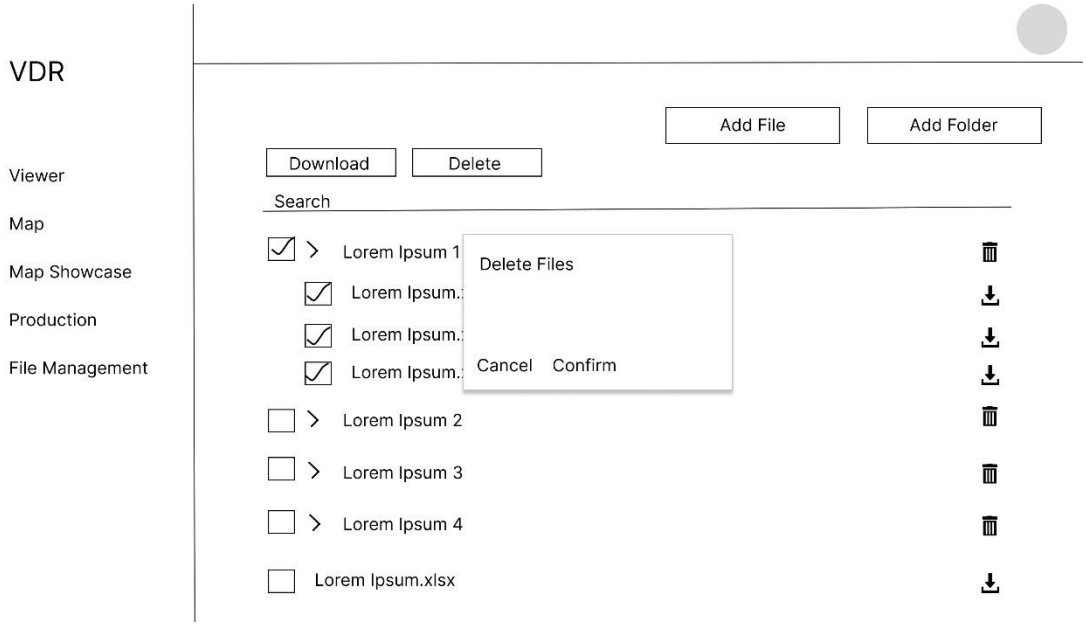


Figure 4.28 File Management Delete Files Design

4.5.5 Admin Page

The admin page as depicted in Figure 4.29, will have the list of users displayed in a table, along with the actions to edit or delete the user.

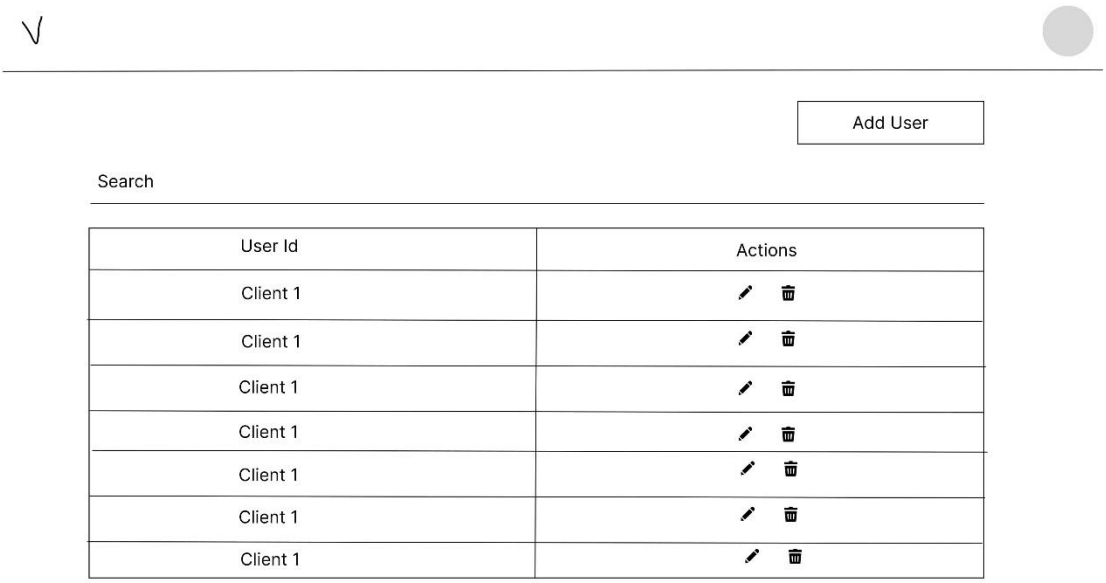


Figure 4.29 Administrator User List Design

To add a new user, the admin can click the ‘Add User’ button, a page as illustrated in Figure 4.30 will appear. There are six columns that the administrator requires to fill in before they can submit which are:

- ‘username’ would be a user identifier and could not be duplicated,
- ‘password’,
- ‘user type’, there are two types, the default type would be ‘Regular User’ and ‘Premium User’,
- ‘name’, and
- ‘affiliation’.

After a user is successfully added, a notification pop up will appear at the bottom of the page as depicted in Figure 4.31. Whereas if the username used already existed, the same notification pop up will show with a message ‘A user with the username already exist’ as illustrated in Figure 4.32.

Figure 4.30 Registering New Client Design

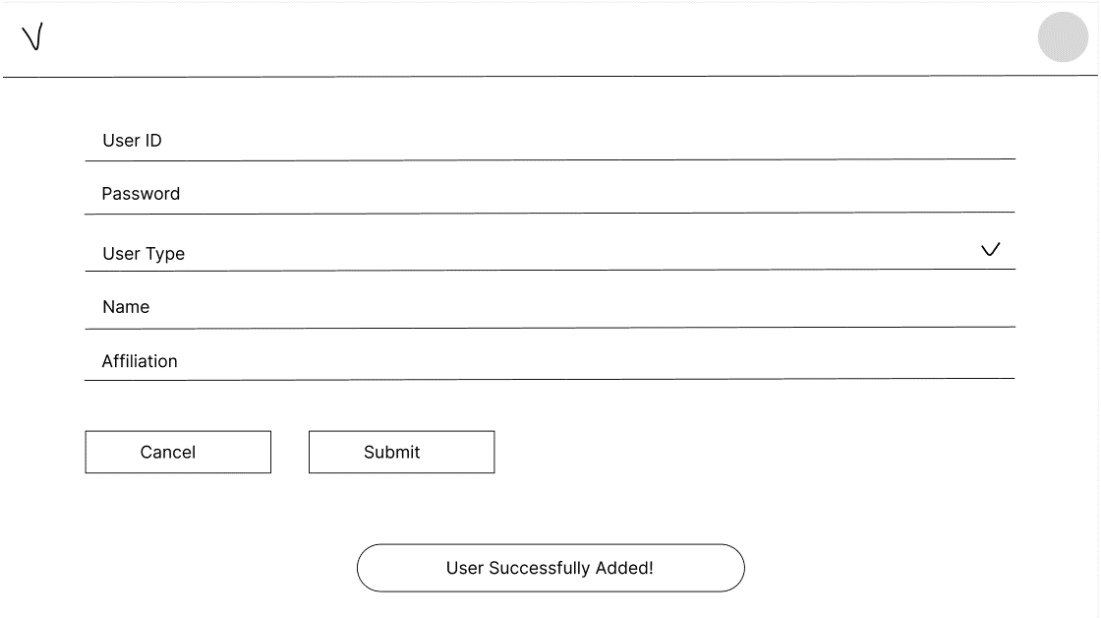


Figure 4.31 Registering New Client Successful Design

Figure 4.32 Registering New Client User ID Duplicate Error Design

When the administrator clicks on the edit icon in Figure 4.29, a page as shown in Figure 4.33 will appear. When editing, the administrator could see the user's profile picture but not change them. In addition, the administrator would be editing all the user detail except for their username. If by any chance the user is having problem with their account password, the administrator would be able to change the password for them as well. After clicking the 'Change Password' button, the page will be changed as illustrated in Figure 4.34. There will be a newly added 'New Password' text column appear, and the 'Change Password' button will be replaced with 'Cancel Change Password' if the administrator wants to cancel the changed password before saving. After the administrator save the changes they made, a notification will appear to indicate the changes has been successful as can be seen in Figure 4.35. As can be seen the difference between Figure 4.30 and Figure 4.33, the expiry date is not user inputted, it is automatically added after the user is added.

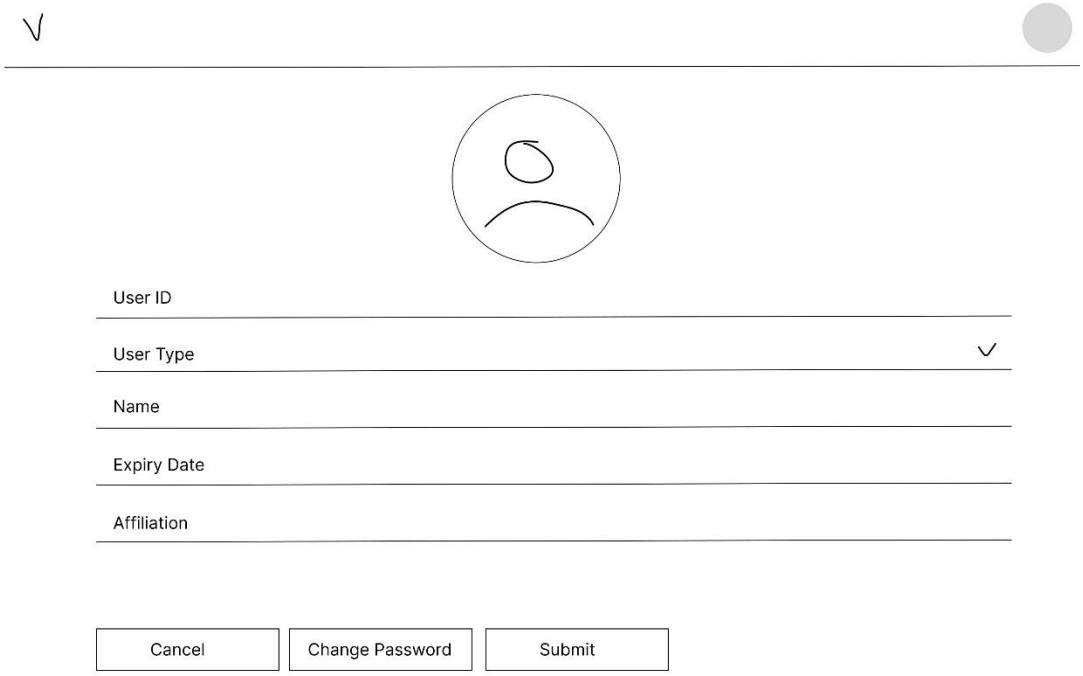


Figure 4.33 Admin Page Edit User Design

Figure 4.34 Admin Page Change User Password Design

Figure 4.35 Admin Page Successful Edit Design

4.5.6 Profile Page

To open the profile page, on the top navbar, there is an avatar icon which is used to display the user's profile picture. When the picture is clicked, a menu will open as illustrated in Figure 4.36. In the menu, the user has the option to go to their profile page or to log out. The user will be redirected to the login page if they choose to log out.

Meanwhile, if the user chooses to go to the profile page, the profile page would display all the user information as displayed in Figure 4.37. The user would not be able to edit their information because name, account expiry date, permission, and their affiliation are a fixed information, if the user would really need to change their information, they need to contact the administrator. However, the user would be able to change their profile picture and password.

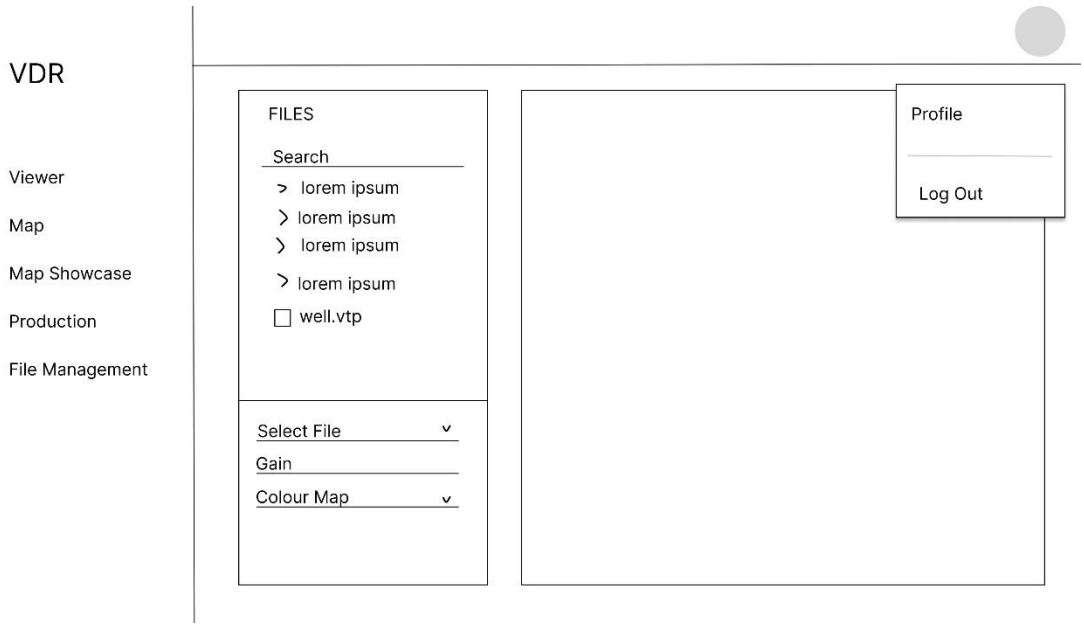


Figure 4.36 Top Bar Profile Menu Design

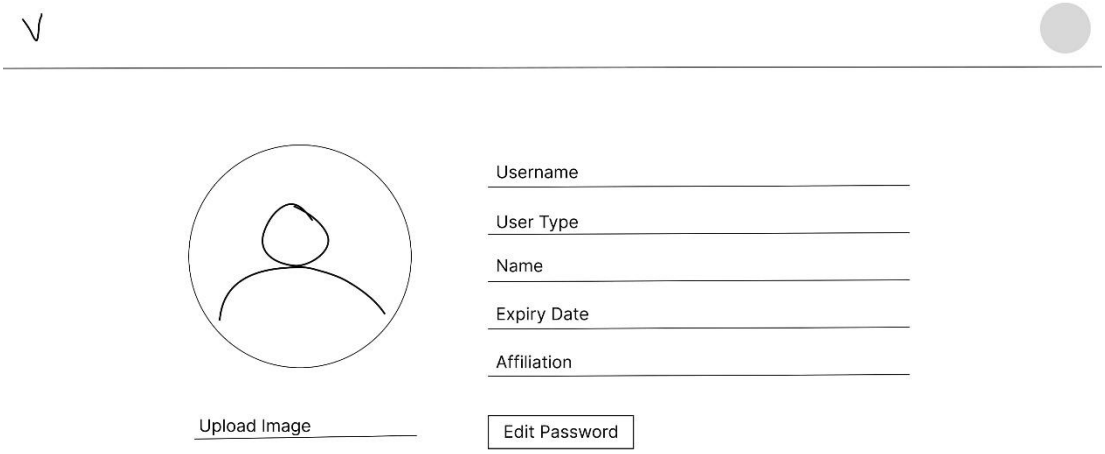


Figure 4.37 Profile Page Design

If the user wants to change their profile picture, they need to click on the ‘Upload Image’ and select their picture. After selecting, the chosen image will be displayed as in Figure 4.38. The user can choose ‘save’ to update their profile picture and cancel to remove the chosen image.

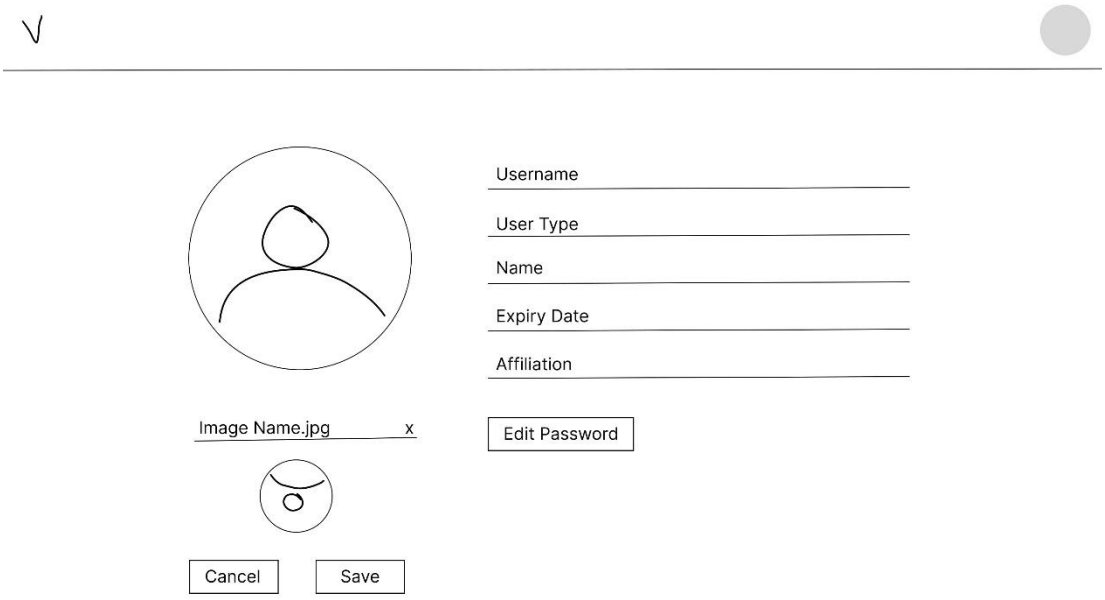


Figure 4.38 Profile Page Change Picture Design

There is a file limit to upload a profile picture, which is 4MB. If the user chooses a picture which exceeds the limit, there will be a warning message as can be seen in Figure 4.39. Moreover, the save button would be disabled and the user only have the option to choose another image or cancel it.

If the save is successful, the preview of new profile picture will disappear directly replace the profile picture.

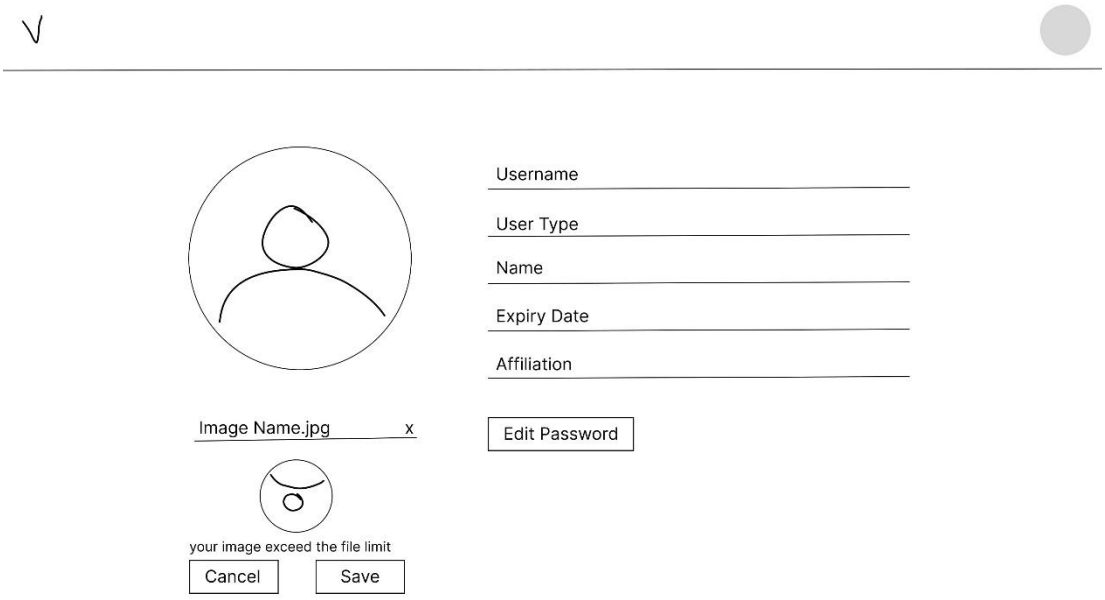


Figure 4.39 Profile Page Image Exceed File Limit Design

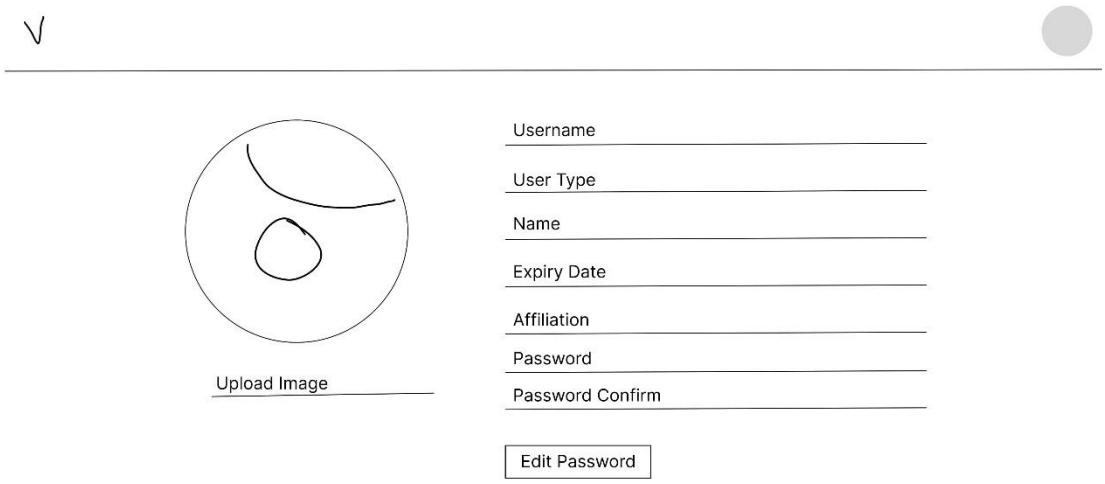


Figure 4.40 Profile Page Edit Password Design

If the user wants to change their password, they need to click the ‘Edit Password’ button and there will be other columns added to enter their new password as illustrated in Figure 4.40. The ‘Edit Password’ button would be removed and there will be an option to cancel or save. Pressing cancel would revert the view back to how it was

before. And a successful save would have a notification pop up as shown in Figure 4.41.

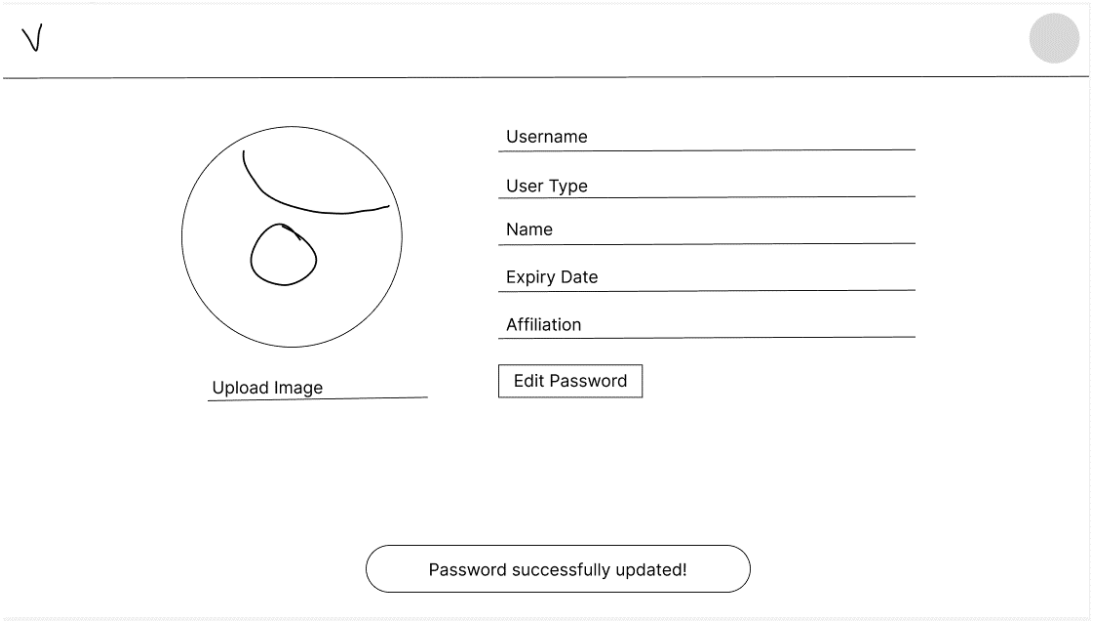


Figure 4.41 Profile Page Successful Password Update Design

CHAPTER 5
IMPLEMENTATION

5.1 State

The state is managed by Vuex state management. The state would store the lists that were fetched from the backend. The state being managed are as described in Table 5.1. The states data will be used and modified accordingly to the actions of the user, however, at reload, the state will return to its initial state.

Table 5.1 State Description

Categories	States	Initial State	Data Type
Authorization	Token	Null	String
	Profile	Null	String
	Logged	False	Bool
	User id	Null	String
	Permission type	Null	String
File	File list	[]	Array
	VTP list	[]	Array
	Sreeya's AI file list	[]	Array
Administrator	User list	[]	Array
Global	Initial load at mount	False	Bool
	Alert tokens expire	False	Bool

The authorization's token and user id are stored to be able to successfully call the APIs. Whereas the permission type is stored to validate whether the user has the authorization to access a certain page or not. The logged state is to check whether or not the user is logged by confirming that a token exists in the local storage. The profile is to store the user's profile picture which is base64 encoded. The list of files and files with pointer specific are stored as the files are going to be used in several features. It is inconvenient for the frontend to fetch the list of files multiple times to accommodate the various features requiring the list of files. Meanwhile, the viewer's list and the user's list are convenient to store as the user does not have to wait for the list to load from the API repetitively when they leave and go back to the page. The global category is to check whether the call to the endpoints is finished, a loading animation would be playing in the case that it's not. While the alert token expire is to check whether the user is automatically logged out due to expiry of the token.

5.2 Route

Routing is managed by Vue in built library the vue-router. Since Vue is building an SPA website and composed of components, Vue Router managed the components into routes, thus, easier for user to access. The routes and components are as described in Table 5.2.

Table 5.2 Routers

Permission	**Routes**	**Components**
Regular User	/vtkviewer	@/views/viewer/VtkViewer.vue
Regular User	/filemanagement	@/views/file-management/Folder.vue

Premium User	/production	@/views/sreeya/Sreeya.vue
Regular User	/not-authorized	@/views/sreeya/NotAuthorized.vue
Public	/login	@/views/pages/Login.vue
Public	/error-404	@/views/Error.vue
Users	/profile/:userid	@/views/profile/Profile.vue
Administrator	/admin	@/views/admin/Admin.vue
Administrator	/admin/adduser	@views/admin/addUser.vue
Administrator	/admin/profile/:userid	@views/admin/editUser.vue

The routes in Table 5.2 all require authentication except for /login. The router will check if the user is logged or not based on the state 'Logged'. If the user is logged, the user will be able to go through the routes permitted to them. Else, the user will be redirected to the login page.

In production page, the application would check through the state 'Permission'. If the permission is a 'Premium User' the user would be able to go through the route, otherwise, they will be redirected to '/not-authorized'. Whereas the premium user is authorized to go through all the routes available for the users. Meanwhile, the administrator does not have any access to what the regular user or the premium user could access, if they tried to access the page, they will be redirected back to /admin.

5.3 Communication with Backend

The communication that the author chooses to use with the backend APIs is through axios. There are four different services which requires the backend APIs, the login, file management, viewer, and the admin. The APIs required for the services are specified

in the Table 5.3. As described in the table, every endpoint would require the access token received from the login to allow the request. The response received from the endpoints described in Table 5.3 are in the form of JSON.

Table 5.3 List of Endpoints

Endpoints	**Methods**	**Description**	**Request**	**Response (200)**
Login - /api/v1/common				

/token	POST	Login for access token	{
			username: string,
			password: string
			}
				{
				"access_token": string
				"token_type": string,
				"type": string
				"name": string
				"expiry_date": string,
				"affiliation": string
				}

File Service - /api/v1/common				

/folders	POST	Create new folder	{
			"paths": [
			"string"
			]
			}
				{
				“status”: “success”
				}

/folders	DELETE	Delete a folder	{
			"paths": [
			"string"
			]
			}
				{
				“status”: “success”
				}

/files	POST	Upload new files	{ files: array[],	{ “status”: “success”

			foldername: string, pointer: string }	}

/files	DELETE	Delete file(s)	{
			"paths": [
			"string"
			]
			}
				{
				“status”: “success”
				}

/files	GET	Download a file	{
			"paths": “string”
			}
				Binary stream

/files/bulk	POST	Download multiple files	{
			"paths": [
			"string"
			]
			}
				Binary stream

/files/lists	GET	Get list of folders and files		{
				“data”: array[
				“id”: string,
				“name”: string,
				“type”: string,
				“children”: array[]
				]
				}

/files/lists/{pointer}	GET	Get list of folders and files based on pointer	{
			“pointer”: string,
			“pathname”: boolean
			}
				{
				“data”: array[
				“id”: int | string,
				“name”: string,

				“type”: string, “children”: array[]] }

/files/lists/	GET	Get list of folders and files based on multiple pointers	{
			“pointer”: string,
			“pathname”: boolean
			}
				{
				“data”: array[
				“id”: int | string,
				“name”: string,
				“type”: string,
				“children”: array[]
				]
				}

Profile Service - /api/v1/registration				

/profile	GET	Get list of users		{
				“data”: array[
				“userid”: string
				]
				}

/profile	PUT	Edit user profile	{
			"userid": string,
			"password": string,
			"type": string,
			"name": string,
			"affiliation": string
			"profile_pict": string
			}
				{
				“status”: “success”
				}

/profile	POST	Add a new user	{
			"userid": string,
			"password": string,
				{
				“status”: “success”
				}

			"type": string, "name": string, "affiliation": string "profile_pict": string }	

/profile/{userid}	GET	Get a user profile	{
			“userid”: string
			}
				{
				"userid": string,
				"password": string,
				"type": string,
				"name": string,
				"affiliation": string
				"profile_pict": string
				}

/profile/{userid}	DELETE	Delete a user	{
			“userid”: string
			}
				{
				“status”: “success”
				}

Machine Learning Service - /api/v1/prediction				

/oil-production	POST	Oil production prediction based on singular data	{
			"hours_online": float,
			"downhole_temp": float,
			"downhole_press": float,
			"press_diff": float,
			"temp_diff": float,
			"deg": string
			}
				{
				"prediction": float
				}

/oil-production/excel	POST	Oil production prediction based on excel file	{
			“path”: string
			}
				{
				“data”: array[
				“label”: string,
				“data”: array[]
				]
				}

/gas-prediction	POST	Gas production prediction based on singular data	{
			"hours_online": float,
			"downhole_temp": float,
			"downhole_press": float,
			"press_diff": float,
			"temp_diff": float,
			"deg": string
			}
				{
				"prediction": float
				}

/gas-prediction-excel	POST	Gas production prediction based on excel file	{
			“path”: string
			}
				{
				“data”: array[
				“label”: string,
				“data”: array[]
				]
				}

5.4 Authentication and Authorization

The authentication and authorization that the application use is the OAuth2 protocol. All the API endpoints used requires the access token. The access token is retrieved when the user logs in and request to the ‘api/v1/common/token’ endpoint. After the user logs in, the endpoint will verify whether the user is authenticated, and if they are, an access token will be returned. After the access token is retrieved, it is both saved in

the state management as can be seen in Table 5.1 and local storage. The access token stored in the local storage is first encrypted before storing. The encryption is using an open-source JavaScript library CryptoJs with their cipher algorithm, the AES. In particular the CryptoJs AES-256, which has a key length of 256 bits.

While the access token stored in the state management is the raw access token which is not encrypted. Since the state management will reset every time the website is reloaded, when the website finished reloading, it will first check the existence of the access token in the local storage. If the encrypted access token in the local storage exists, it will decrypt the encryption. The decrypted access token will then be stored in the state management. Else, the website will redirect the user to the login page.

The access token is set to a time limit for a session after it is retrieved. Thus, the frontend side will check for any error regarding authentication. The error code set for token expire or false authentication is 401 and 403. When the response received from the endpoints is an error, provided that the error status code is 401 or 403, the page will immediately be redirected to the login page so the user could ask for another access token.

5.5 User Interface

The following will consist of screenshots from the website application running after connected to the APIs.

5.5.1 Login Page

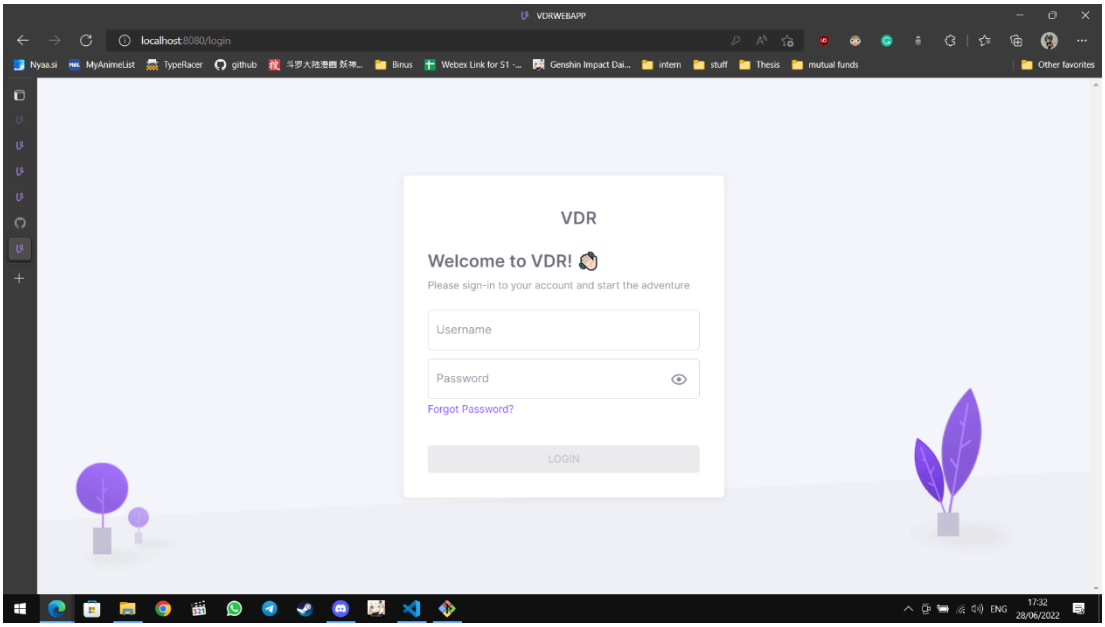


Figure 5.1 Login Page

After the user typed their username and password and press the login button, the user will have to wait while the application loads their data.

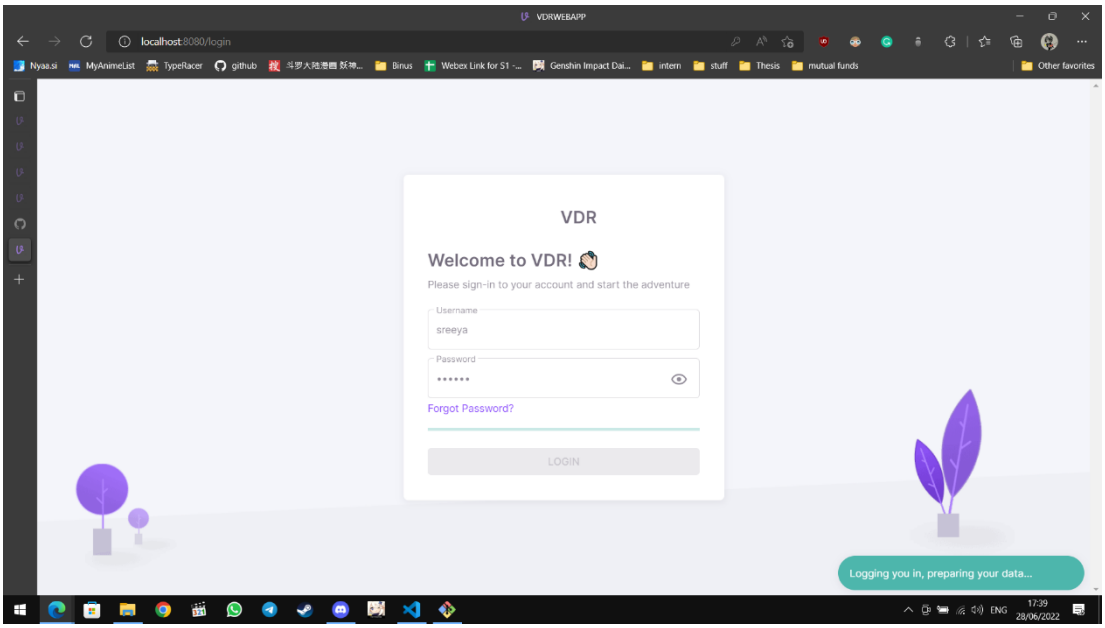


Figure 5.2 Login Page Loading

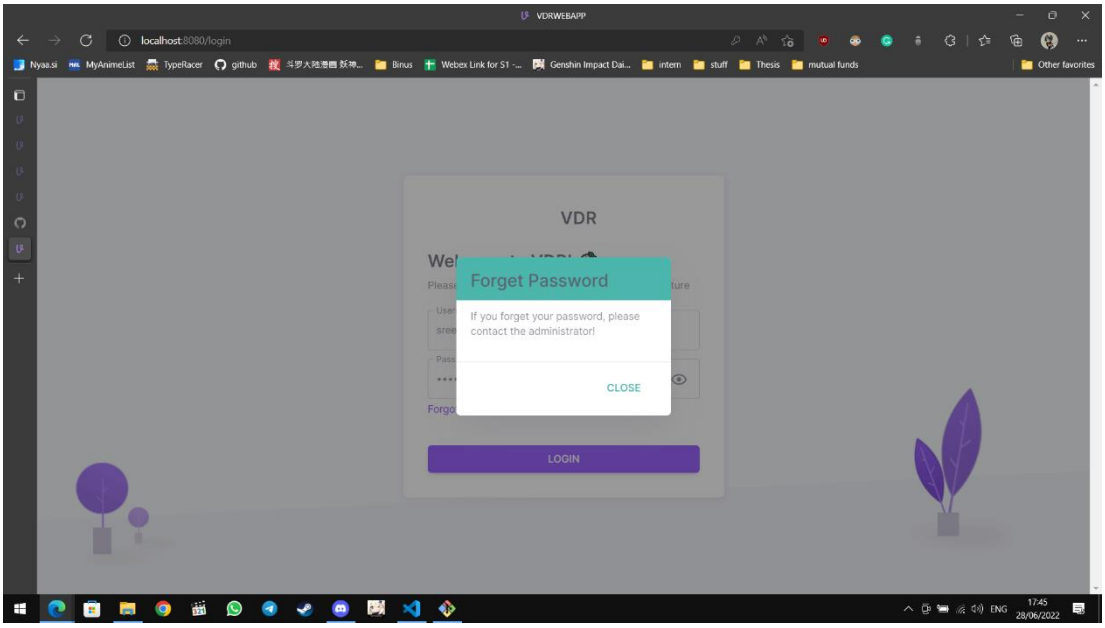


Figure 5.3 Login Page Forget Password

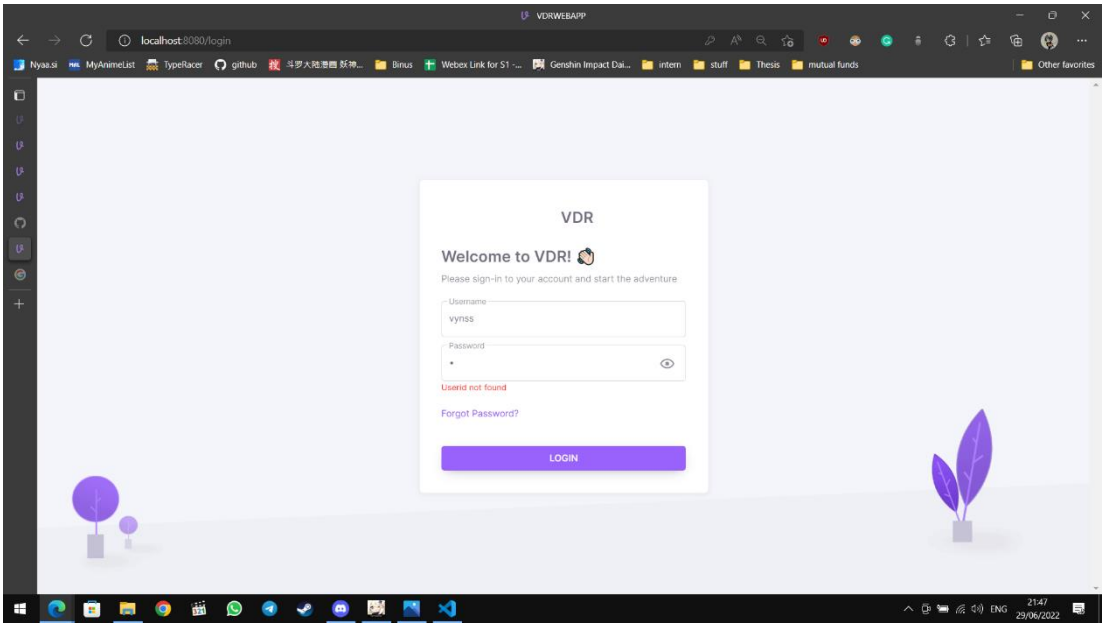


Figure 5.4 Login Page Username Not Found

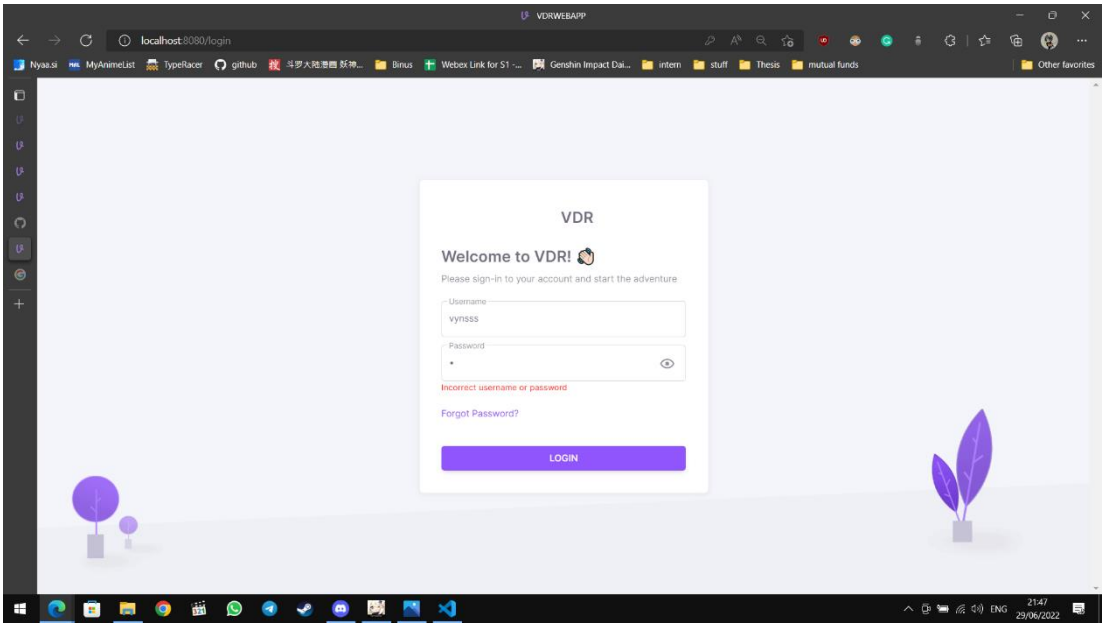


Figure 5.5 Profile Page Incorrect Password

5.5.2 Viewer Page

As depicted in Figure 5.6, the list of files and folders are loaded by the state ‘VTP list’. The pointer used to display the files and folder is ‘well-vtp’, ‘surface-vtp’, and ‘line-vtp’. There’s also an additional button to add .vtp files which will redirect to VTP Form page illustrated in Figure A. 5. To display the .vtp file which is shown in Figure 5.7, an open-source library VTKjs is used.

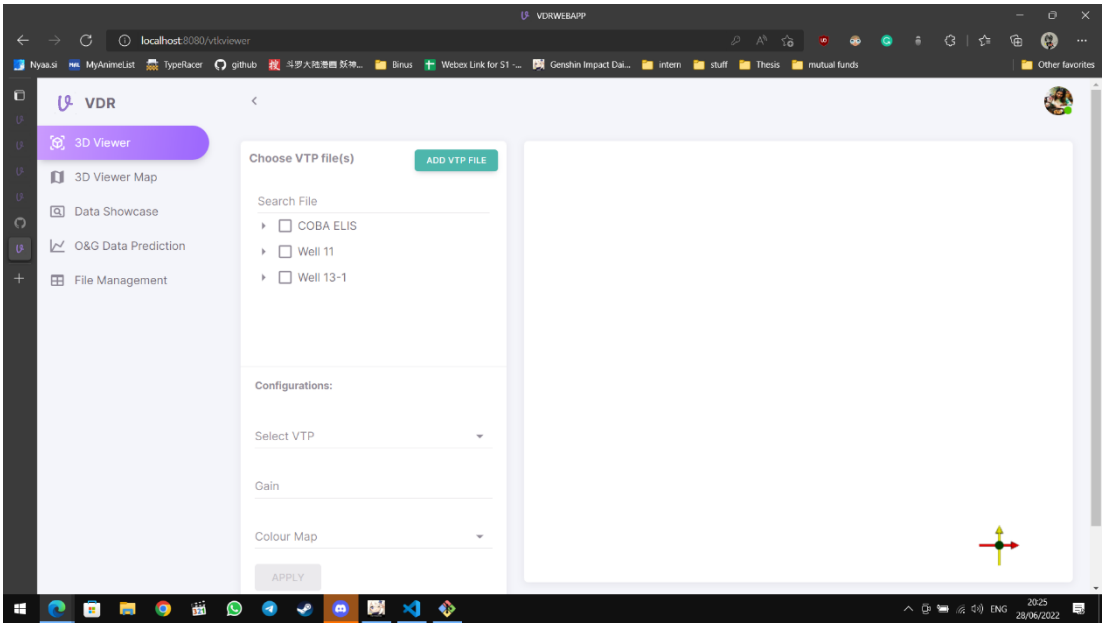


Figure 5.6 3D Viewer Page

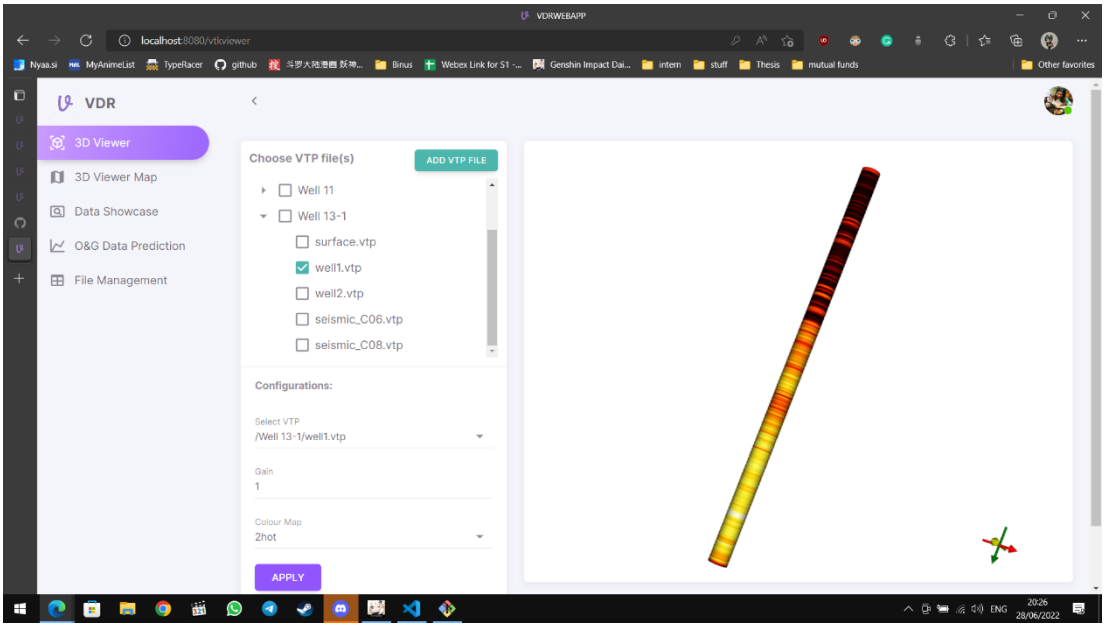


Figure 5.7 3D Viewer Page Select File

After a user select a file from the treeview, it will be as shown in Figure 5.7, showing the .vtp file interactively. Meanwhile, if the user chose multiple files, it will be as depicted in Figure 5.8.

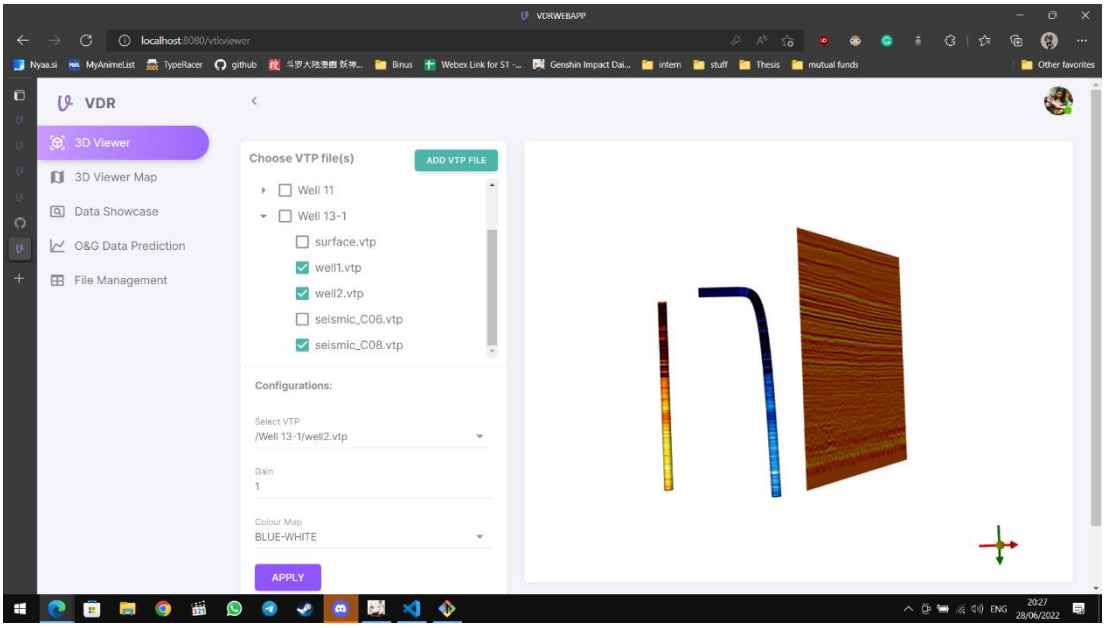


Figure 5.8 3D Viewer Page Multi Select Files

5.5.3 Production Page

The list of files and folders shown in Figure 5.10 is using pointer ‘production’ which is stored in ‘Sreeya AI’s’ list. In the case that the list is empty, the page will be as shown in Figure 5.9.

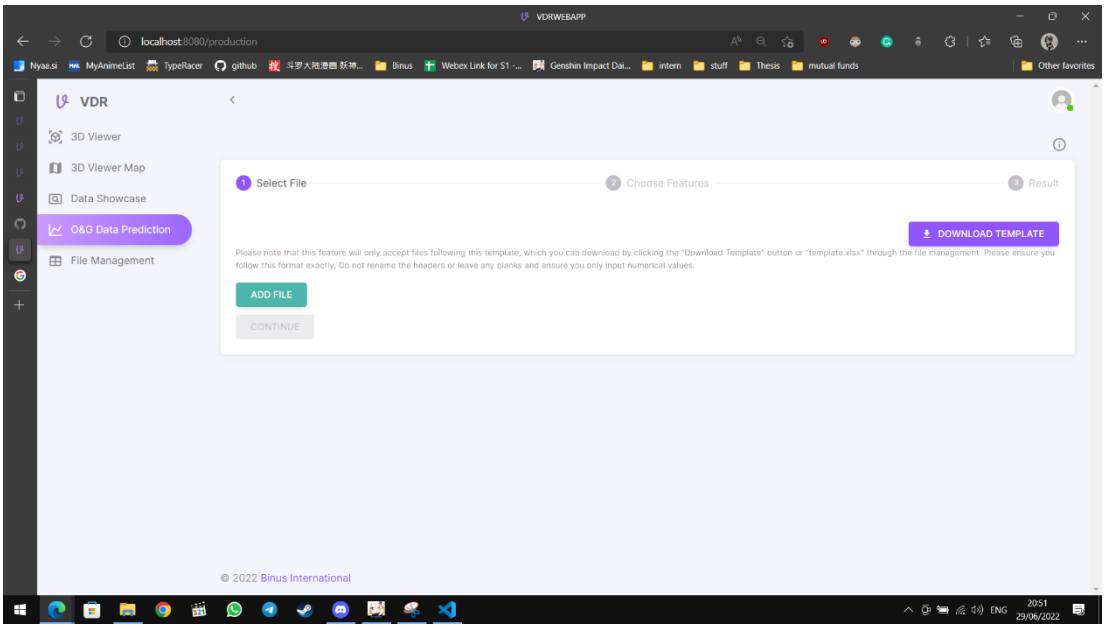


Figure 5.9 Production Page Empty List

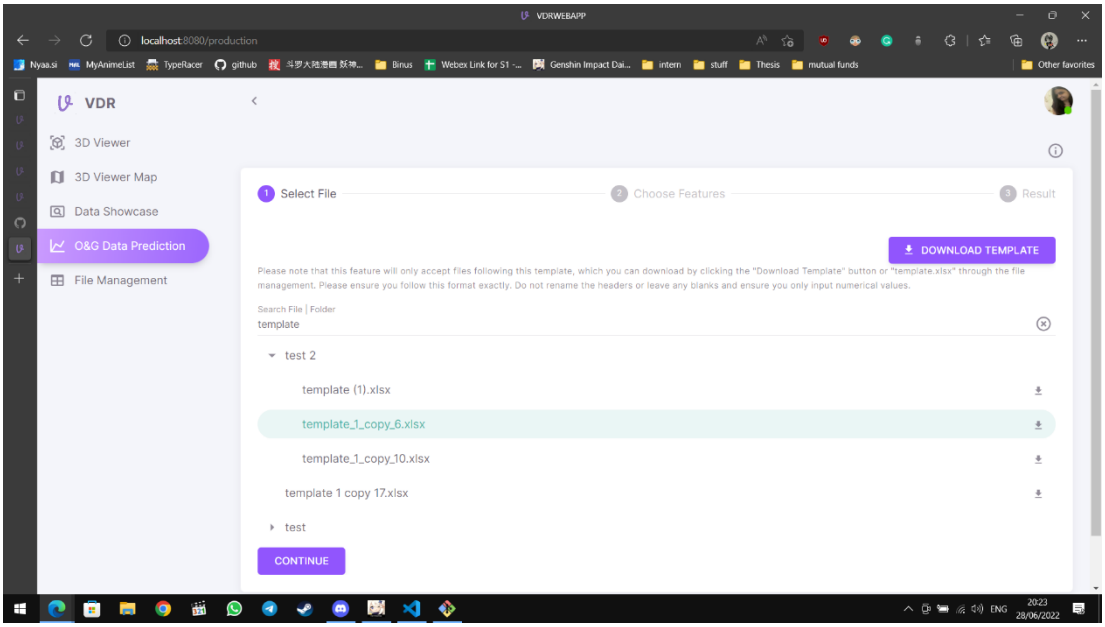


Figure 5.10 Production Page Step 1

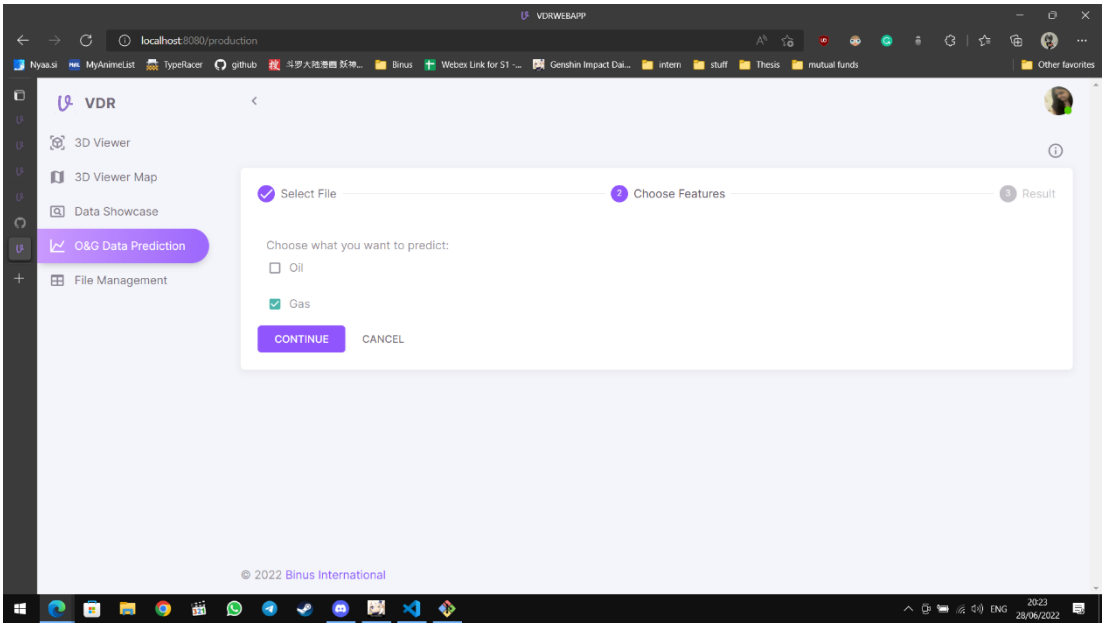


Figure 5.11 Production Page Step 2

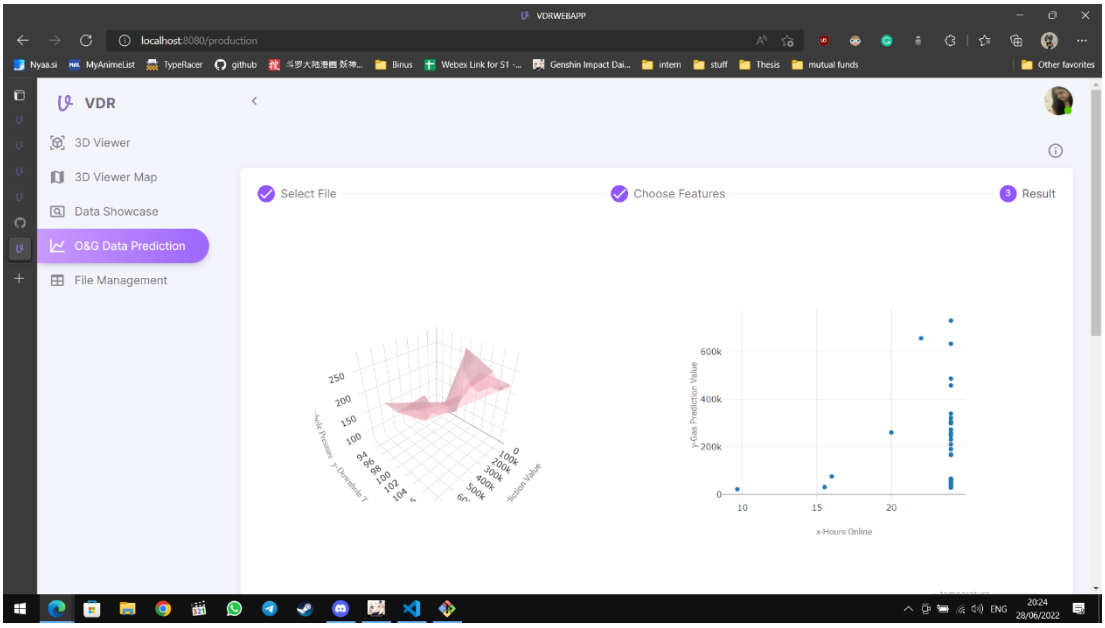


Figure 5.12 Production Page Step 3 Part 1

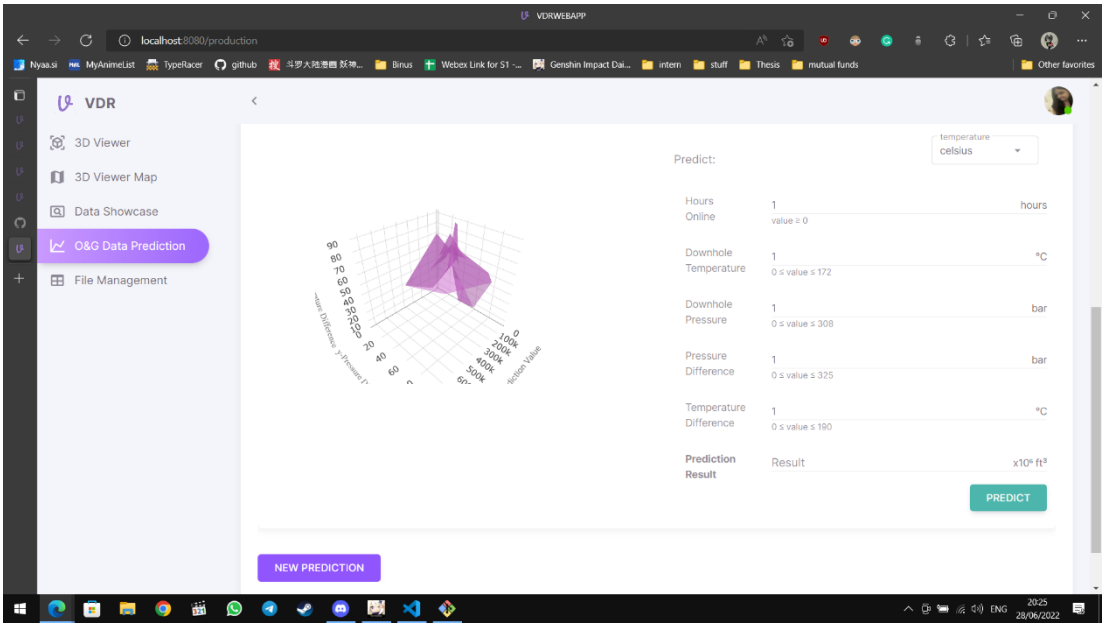


Figure 5.13 Production Page Step 3 Part 2

In Figure 5.13, for the additional prediction, the user would be able to choose the temperature measurement, the default measurement would be ‘celsius’, however, they are able to change the measurement to ‘fahrenheit’. There is also a range of values for each field, if the user input a number that is out of the range, the ‘predict’ button would be disabled. In the case that there is an error with the file, or if the file selected is not

according to the template in Figure 4.16, which will return an error code 400 or 422, it will be shown as illustrated in Figure 5.14. The ‘New Prediction’ button will redirect them to step 1.

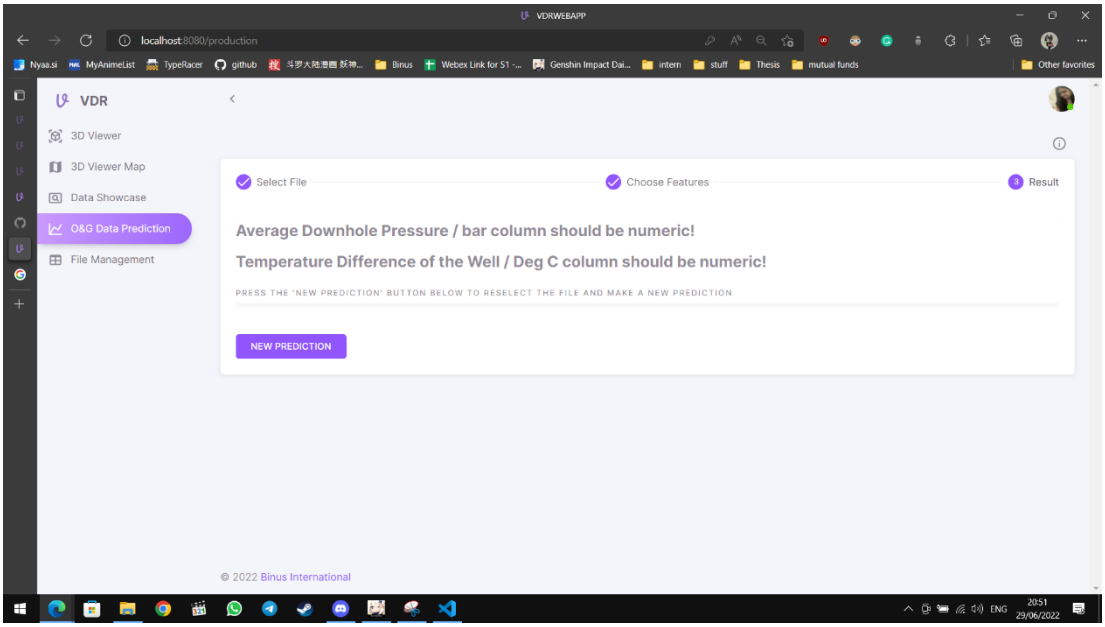


Figure 5.14 Production Page File Error

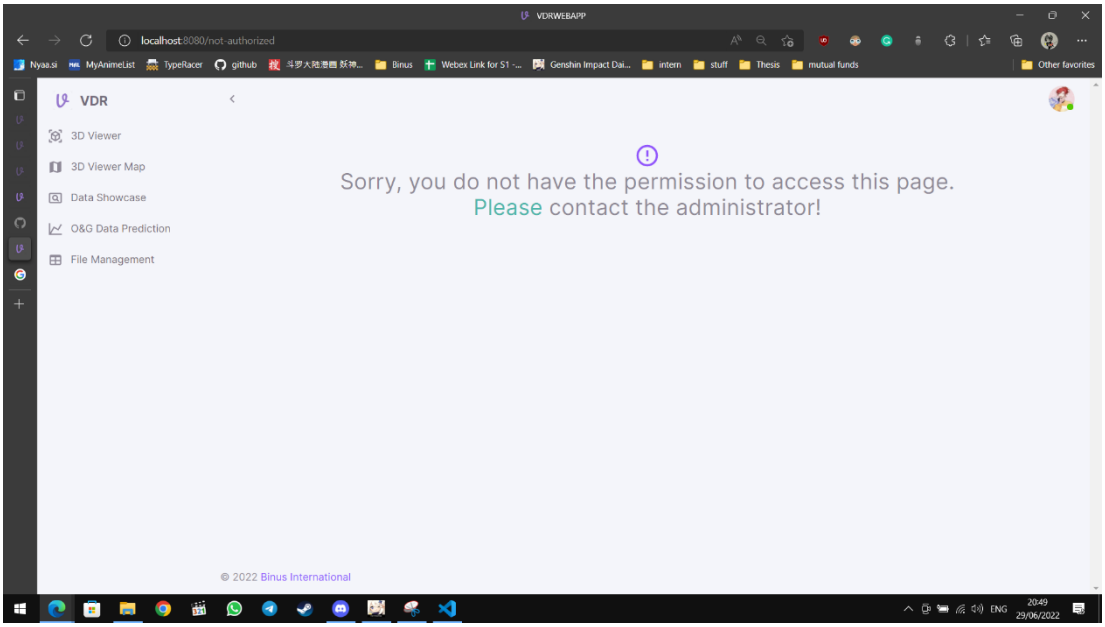


Figure 5.15 Production Page Not Authorized

5.5.4 File Management Page

The list of files and folders as depicted in Figure 5.16 is from the state ‘File list’.

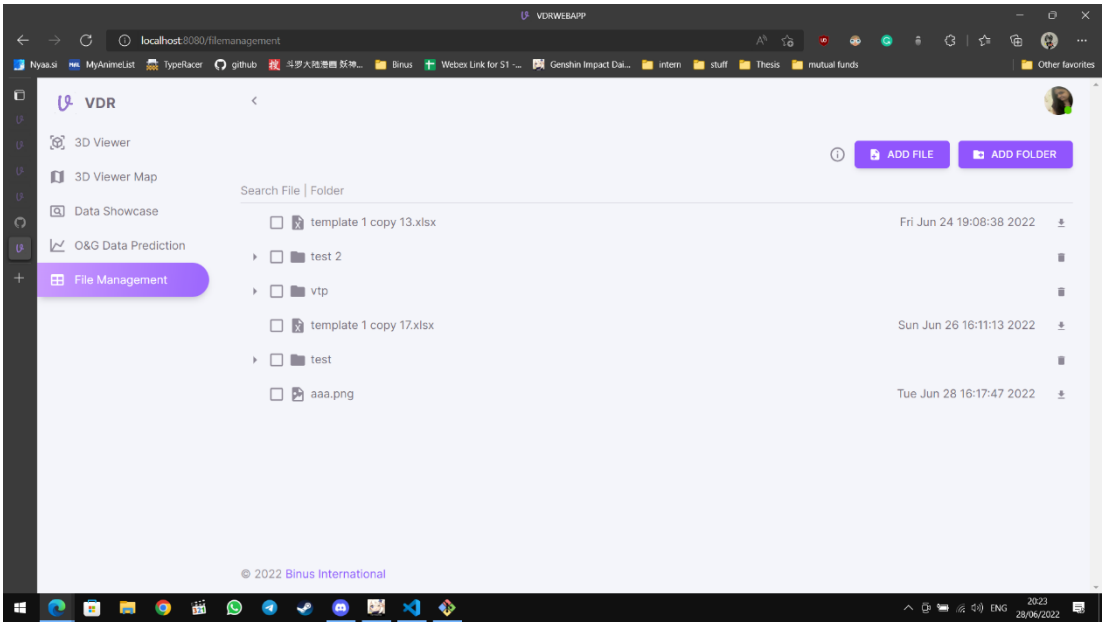


Figure 5.16 File Management Page

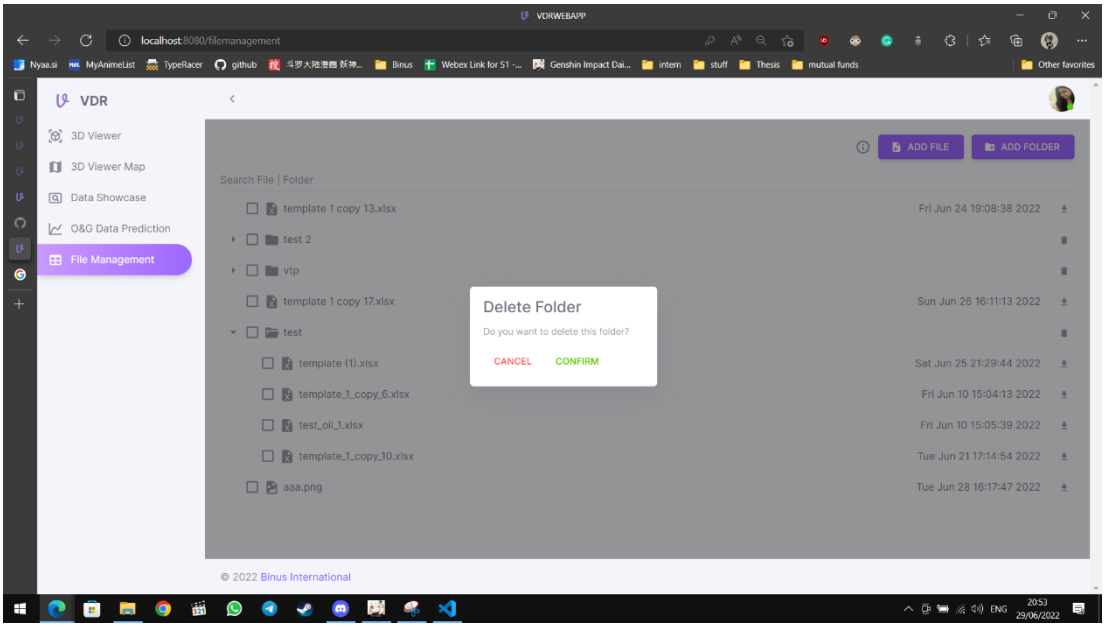


Figure 5.17 File Management Page Delete Folder

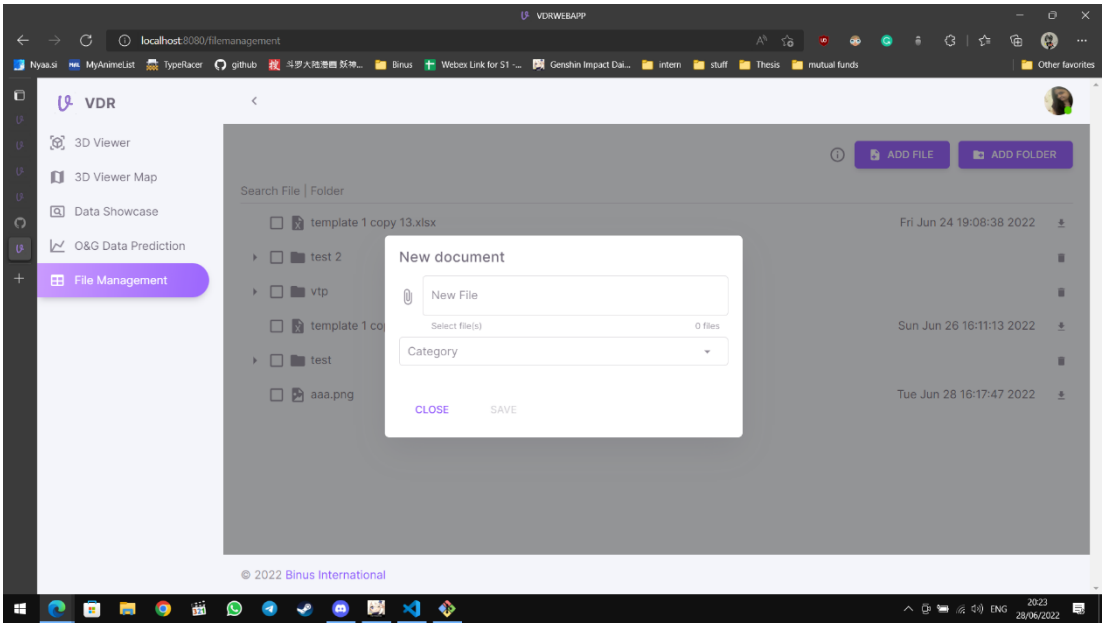


Figure 5.18 File Management Page Add File

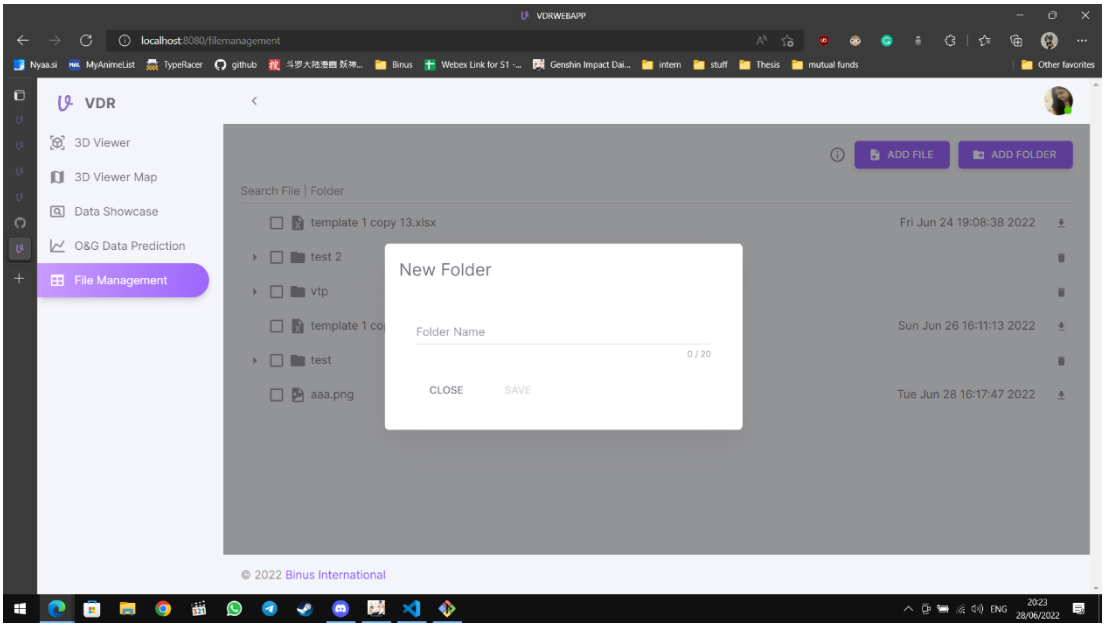


Figure 5.19 File Management Page Add Folder

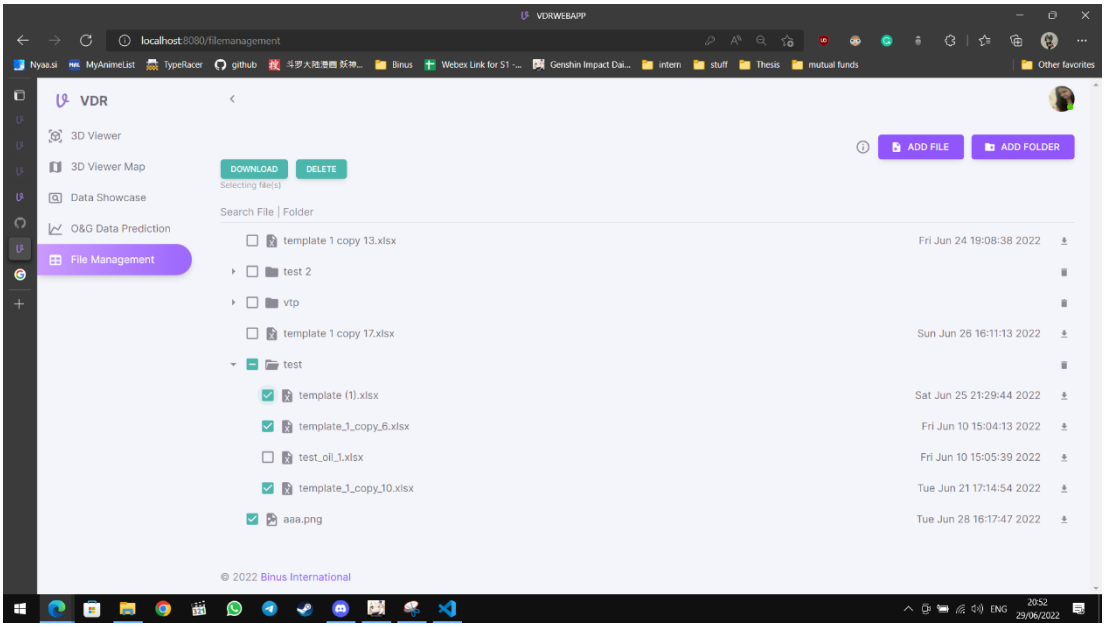


Figure 5.20 File Management Page Checkbox

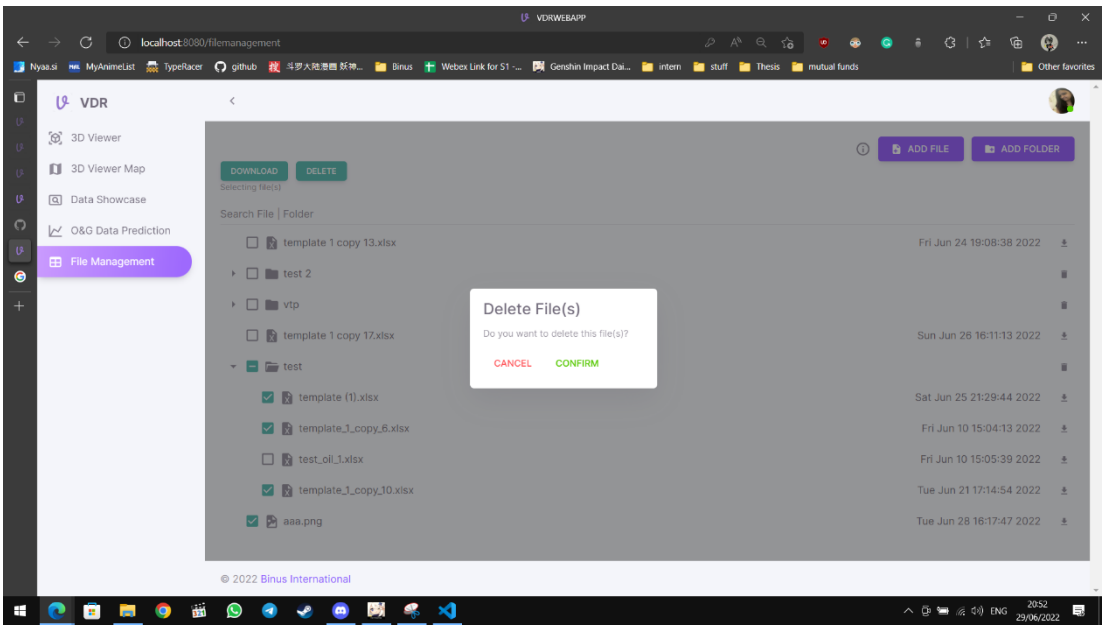


Figure 5.21 File Management Page Delete Files

5.5.5 Admin Page

The list of users is obtained from the state ‘User list’.

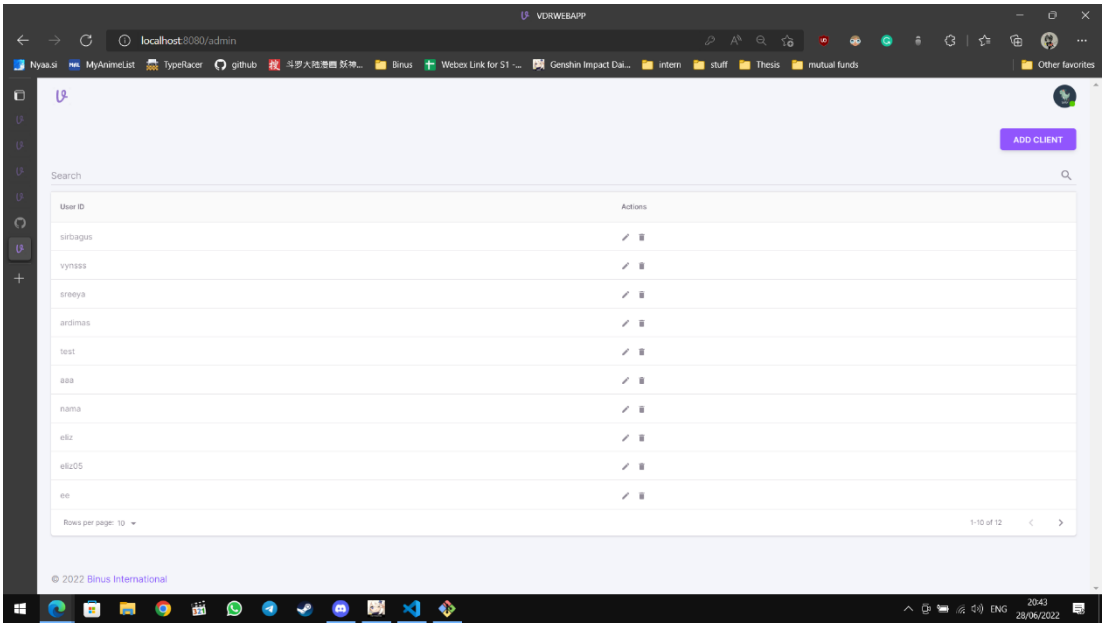


Figure 5.22 Administrator Page

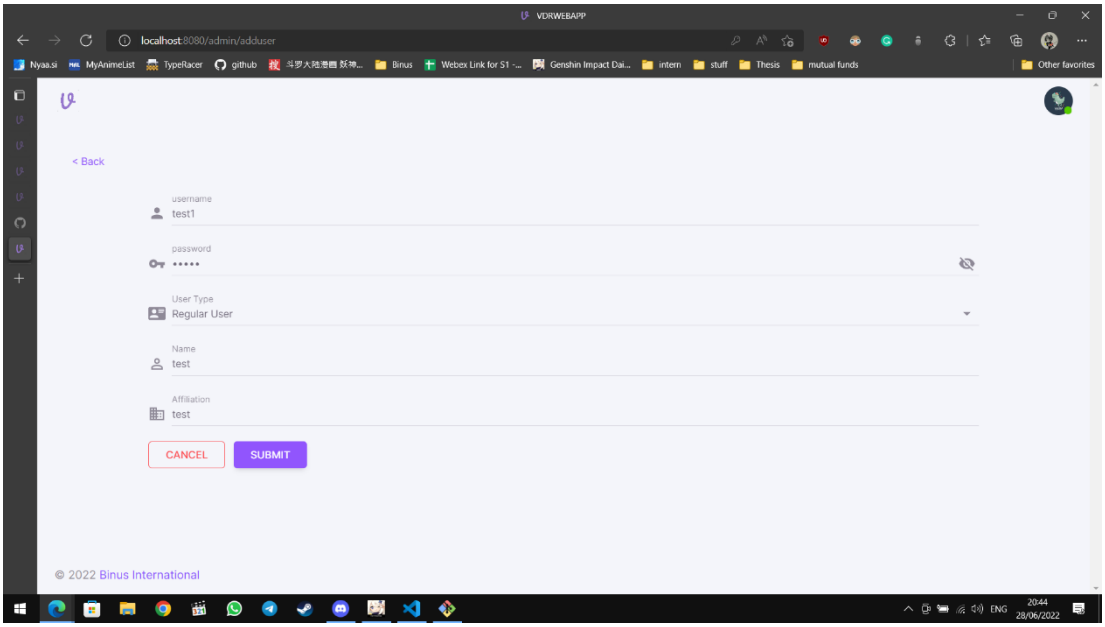


Figure 5.23 Administrator Page Add User

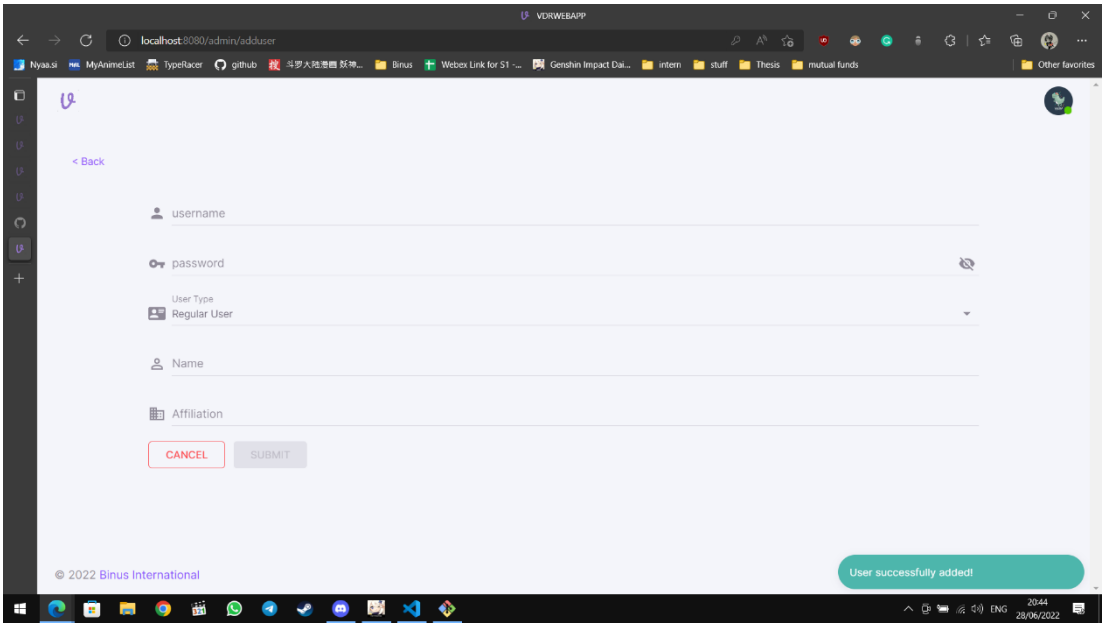


Figure 5.24 Administrator Page Successfully Added

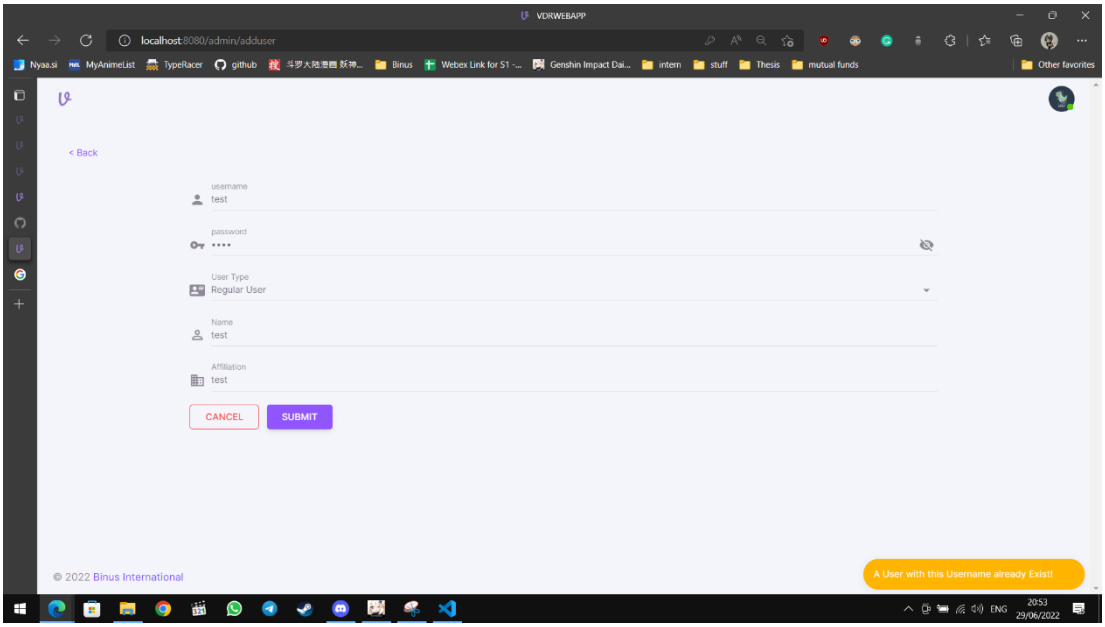


Figure 5.25 Administrator Page Add User Duplicate Username

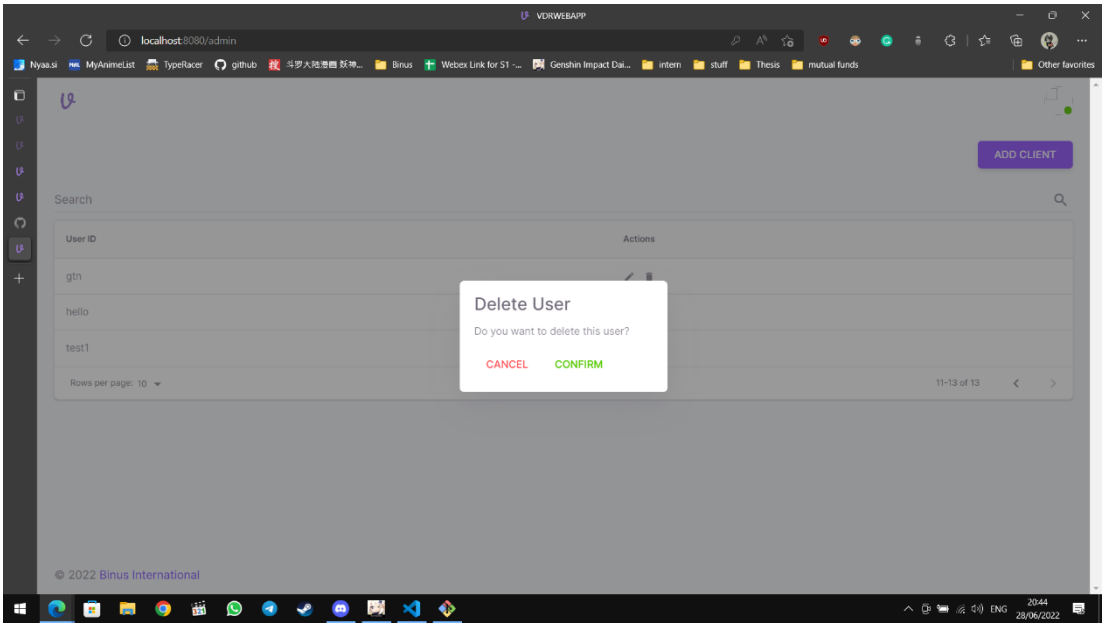


Figure 5.26 Administrator Page Delete User Confirmation

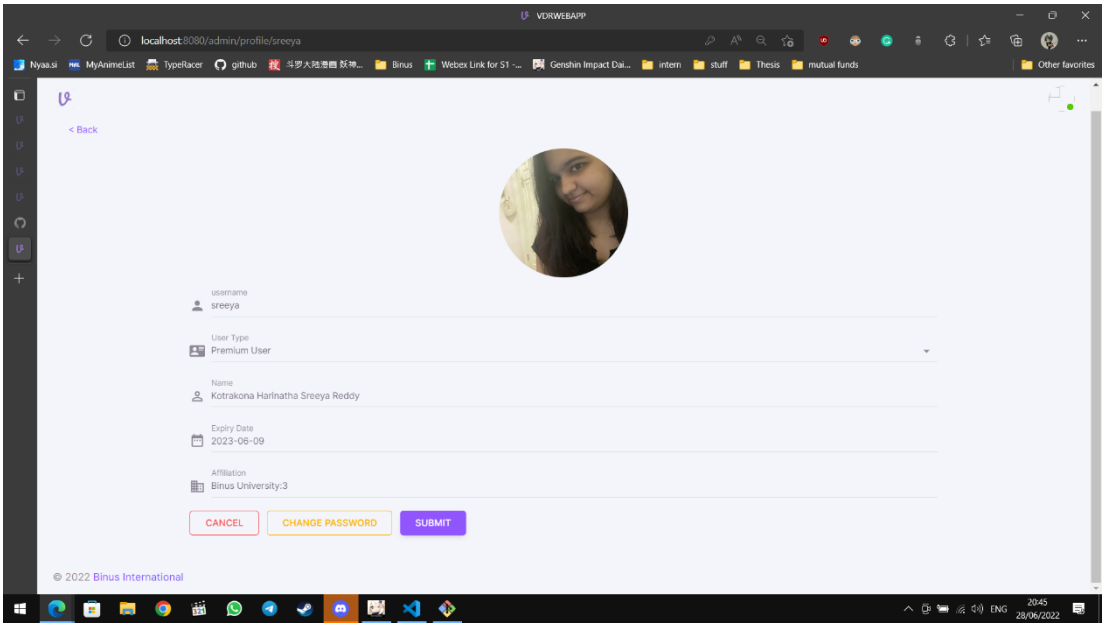


Figure 5.27 Administrator Page Edit User

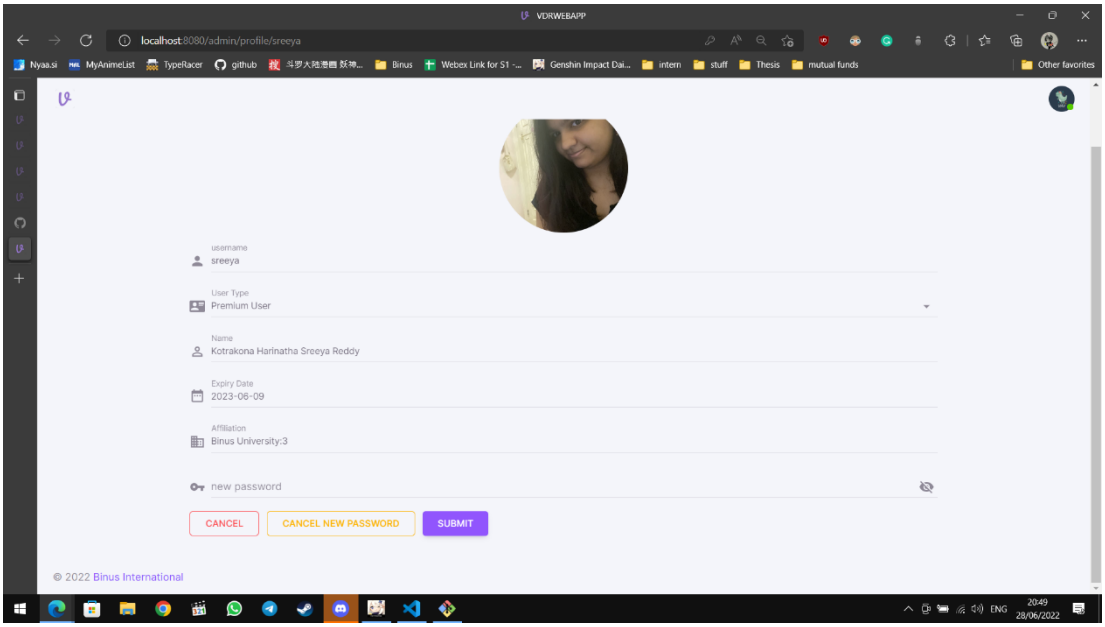


Figure 5.28 Administrator Page Edit User Password

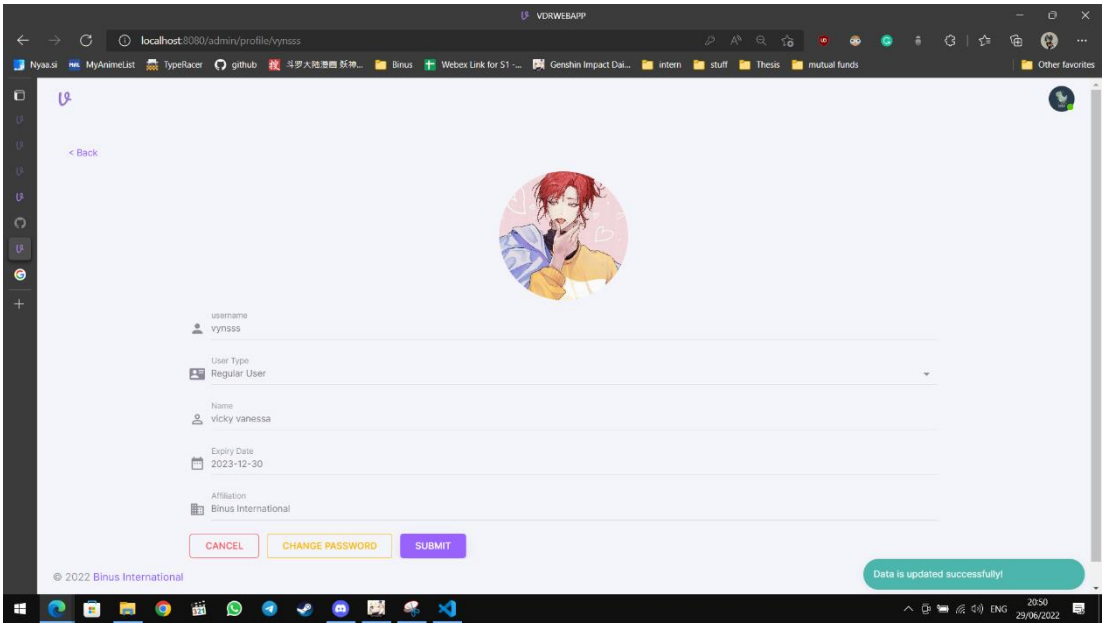


Figure 5.29 Administrator Page Edit User Successfully

5.5.6 Profile Page

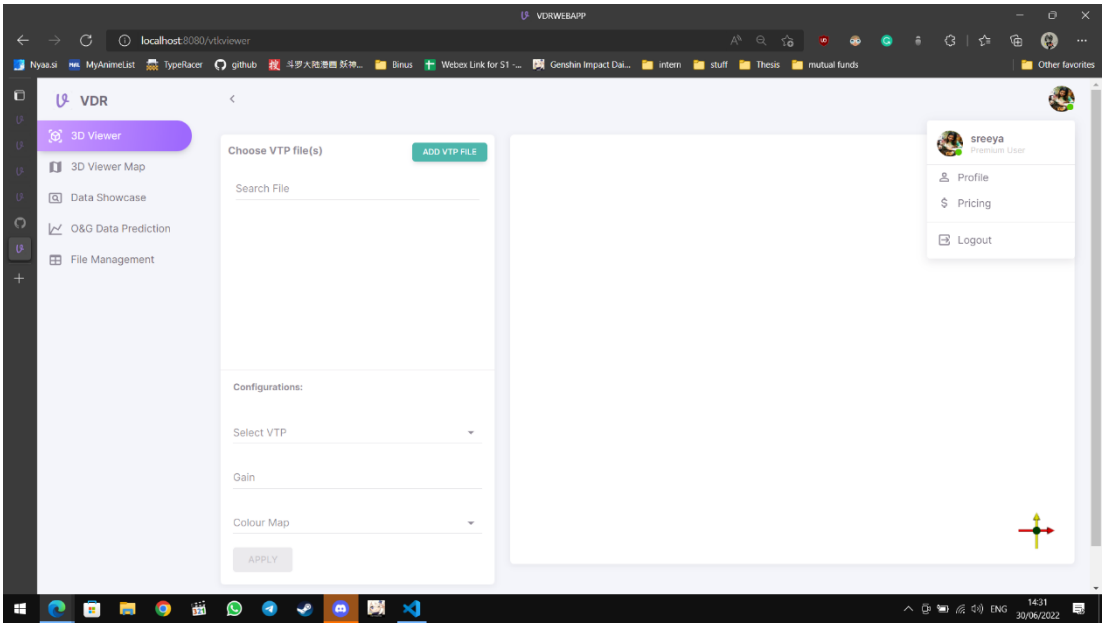


Figure 5.30 Top Bar Profile Menu Page

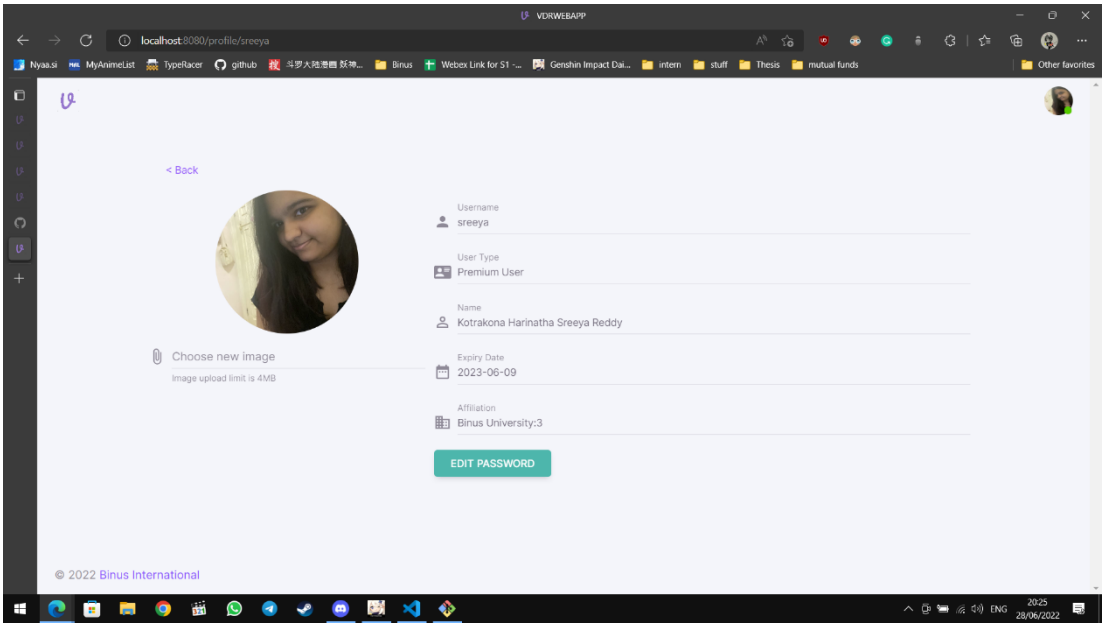


Figure 5.31 Profile Page

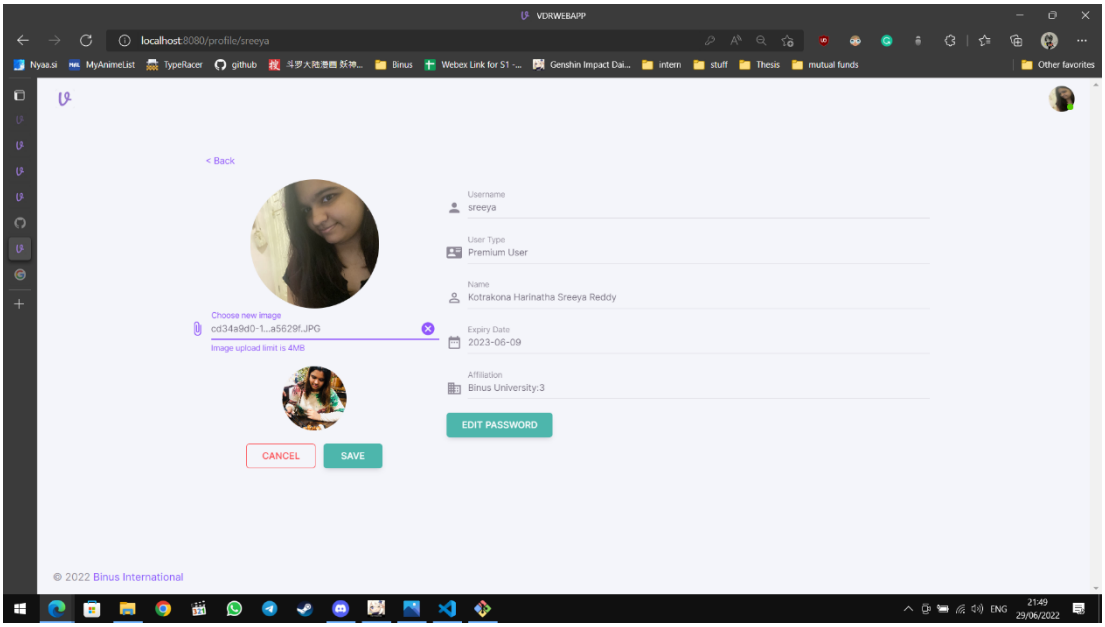


Figure 5.32 Profile Page Upload New Profile Picture

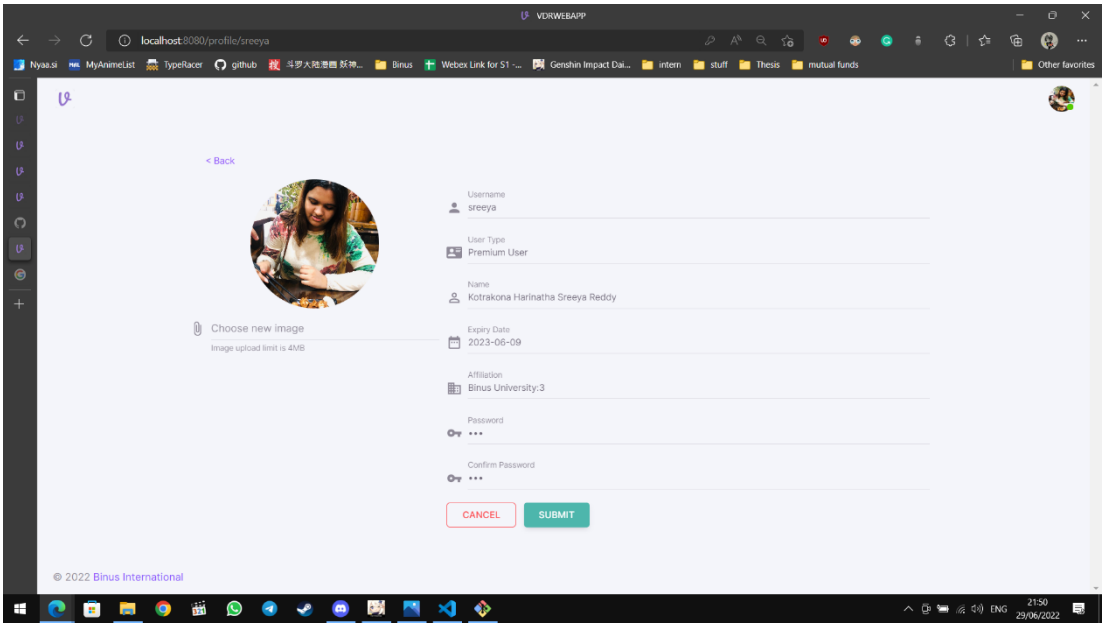


Figure 5.33 Profile Page Change Password

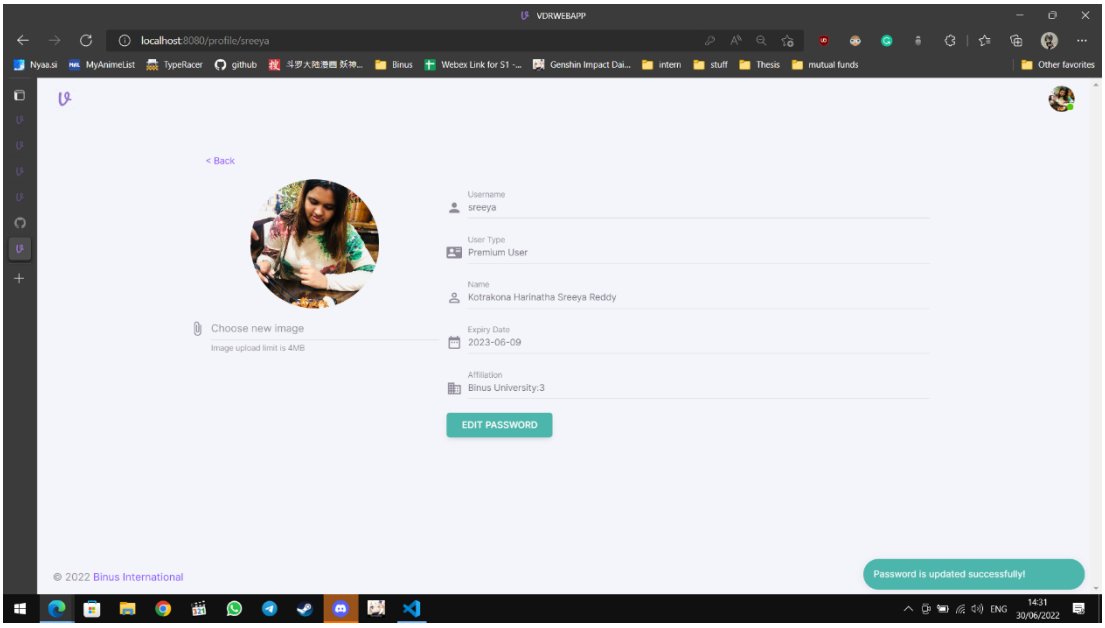


Figure 5.34 Profile Page Change Password Successful

5.5.7 Miscellaneous Page

Figure 5.35 shows what the website will show it the user choose a router that does not exist in the router.

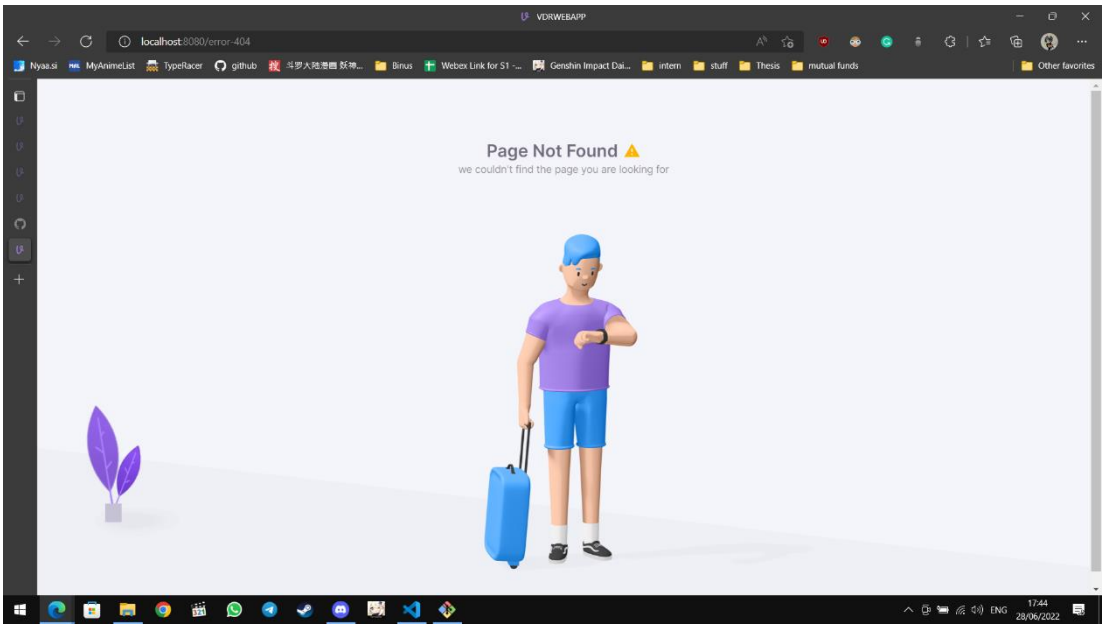


Figure 5.35 Error 404 Page

CHAPTER 6
DISCUSSION

6.1 Development

During the development which span over 4 months, the frontend VDR website application has experienced several changes to what the author's team considered would accommodate the client and user. The main features of the website application are as discussed in the scope of work stated in Figure A. 4, the frontend has been modified from the initial design and redesigned repeatedly until it is how it looks by the end of the development. Moreover, by using OSS, the cost of development of the VDR website application is negligible.

6.2 Evaluation

The author's team are not able to do a UI testing during the development of the website as there are limits to the duration of the development. However, although a UI testing is not able to be conducted, the author's team have presented and did a demonstration to both the product owner and the clients. The result is they are satisfied with the progress and the UI, the proof can be seen in **Transcript?!**

As this development is only on the first phase, there could be major improvement for the latter phases to completely develop a website application that could aid the whole oil and gas industry in Indonesia.

6.3 Constraints

The constrains that the author's team experienced in developing the VDR website application has by the end of the development are:

- Responsiveness

  The author had tried their best to make the website application as responsive as possible. However, there are some big components on the website which could not be made responsive.

- Testing

  Due to the author focusing on the functionality of the features, by the end of the development, the author still lacked the time to do a proper testing on the application

- Real application reference

  Most of oil and gas application are paid, thus, the author and the team does not have the real application reference to fulfill the user needs and could only rely on the client. A full demonstration and exploration could lead the author's team to develop a better UI to better accommodate the user.

CHAPTER 7
CONCLUSION AND RECOMMENDATION

7.1 Conclusion

At the end of the development, the author concluded that client has agreed that the website application fulfills the scope given to the author's team and the prototype requirements. Moreover, the author's team accomplished the aims stated in Section 1.3 which are:

- building a cheaper VDR website application with the aid of OSS,
- a predictive model which is capable of predicting the oil and gas production, and
- visualizing the oil and gas prediction data in the production page and the well log, seismic, and surface data.

By accomplishing the aims, the author's team also fulfills the benefit that the VDR website application aim to provides.

7.2 Recommendations

Based on what the author experienced during development and communication with the client, these are some recommendations that could improve the VDR website application in the future:

- In the Viewer page, in the case that there are hundreds or thousands of files, instead of ticking the checkboxes to show the .vtp files, a slider of the files could be made to increase the responsiveness of the view and changes of the file.

- Although the frontend apply encryption to the access token, it would be preferred if the backend sends an encrypted token so the raw access token would be completely hidden from the client side.
- Instead of asking the user to contact the administrator repeatedly in case there are some complications on the user side, perchance building a direct chat feature to contact the administrator would be better to accommodate both the user and administrator.

REFERENCES

[1] M. S. Vassiliou, Historical Dictionary of the Petroleum Industry, Rowman & Littlefield, 2018.

[2] W. Faisol, S. Indriastuti and A. Trihartono, "Indonesia and OPEC: Why does Indonesia maintain its distance?," *IOP Conference Series: Earth and Environmental Science*, vol. 485, p. 012010, 2020.

[3] I. S. Chandranegara. and Z. A. Hoesein, "Policy concept and designs of oil and gas governance in Indonesia's oil companies," *International Journal of Energy Economics and Policy*, vol. 9(3), pp. 121-127, 2019.

[4] B. Harrison, "The Data Room," *Developments in Petroleum Science*, vol. 69, pp. 21-26, 1 October 2020.

[5] "Intviewer - Fast Geoscience Visualization, Analysis & QC," INT, 02 August 2021. [Online]. Available: https://www.int.com/products/intviewer/.

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

[7] Geodwipa Teknika Nusantara, "ptgtn," GTN, [Online]. Available: https://ptgtn.com/.

[8] Schlumberger, ProSource Front Office User Guide, Texas, 2013.

[9] "ProSource front office," ProSource, [Online]. Available: https://www.software.slb.com/products/prosource/prosource-front-office#sectionFullWidthTable.

[10] "Secure, Remote Access to Field Datasets Enables Potential Investors to Complete Asset Evaluations," Schlumberger, [Online]. Available: https://www.slb.com/resource-library/case-study/dss/delfi-virtual-data-room-generic-asia-pacific-cs.

[11] M. Lewis, "The Importance of a Virtual Data Room Index and Folder Structure," Deal Room, [Online]. Available: https://dealroom.net/blog/the-importance-of-a-virtual-data-room-index-and-folder-structure.

[12] The Iowa Institute for Biomedical Imaging, "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..

[13] vtk, "VTK - The Visualization Toolkit," [Online]. Available: https://vtk.org/.

[14] M. Jazayeri, "Some Trends in Web Application Development," *Future of Software Engineering (FOSE '07)*, pp. 199-213, 2007.

[15] J. Tie, J. Jin and X. Wang, "Study on application model of three-tiered architecture," in *Second International Conference on Mechanic Automation and Control Engineering*, Inner Mongolia, 2011.

[16] Y. Xing, J. Huang and Y. Lai, "Research and analysis of the front-end frameworks and libraries in E-Business Development," *Proceedings of the 2019 11th International Conference on Computer and Automation Engineering - ICCAE 2019*, 2019.

[17] N. Li and B. Zhang, "The research on Single Page Application Front-end development based on Vue," *Journal of Physics: Conference Series,* vol. 1883, 2021.

[18] N.-A. Sireteanu and D. Homocianu, "Front-end frameworks for development of spa and MPA web application," *SSRN Electronic Journal,* December 2021.

[19] V. Hutagikar and V. Hegde, "Analysis of Front-end Frameworks for Web Applications," *International Research Journal of Engineering and Technology (IRJET),* vol. 07, no. 04, April 2020.

[20] "Getting started - vue.js," Vue.js, [Online]. Available: https://012.vuejs.org/guide/.

[21] D. Saha and A. Mandal, "User Interface Design Issues for Easy and Efficient Human Computer Interaction: An Explanatory Approach," *INTERNATIONAL JOURNAL OF COMPUTER SCIENCES AND ENGINEERING,* vol. 3, no. 1, pp. 127-135, 2015.

[22] Interaction Design Foundation, "What is User Interface Design?," Interaction Design Foundation, [Online]. Available: https://www.interaction-design.org/literature/topics/ui-design.

[23] wix, "What is User Interface (UI)?," wix, [Online]. Available: https://www.wix.com/encyclopedia/definition/user-interface-ui.

[24] microsoft, "Following User Interface Guidelines - Win32 apps," 24 August 2019. [Online]. Available: https://docs.microsoft.com/en-us/windows/win32/appuistart/following-guidelines.

[25] Google, "Material Design," [Online]. Available: https://material.io/.

[26] "Material Components," [Online]. Available: https://github.com/material-components/material-components.

[27] Vuetify, "Why Vuetify," [Online]. Available: https://vuetifyjs.com/en/introduction/why-vuetify/#why-vuetify3f.

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

[29] W3 schools, "What is JSON?," w3school, [Online]. Available: https://www.w3schools.com/whatis/whatis_json.asp.

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

[31] geeks for geeks, "Difference between Authentication and Authorization," Geeks for Geeks, 31 May 2022. [Online]. Available: https://www.geeksforgeeks.org/difference-between-authentication-and-authorization/.

[32] Auth0, "Authentication vs. Authorization," [Online]. Available: https://auth0.com/docs/get-started/identity-fundamentals/authentication-and-authorization.

[33] Auth0, "Glossary - Access Token," [Online]. Available: https://auth0.com/docs/glossary?term=access-token.

[34] "Types of Authentication Protocols," Geeks for Geeks, 17 January 2021. [Online]. Available: https://www.geeksforgeeks.org/types-of-authentication-protocols/.

[35] geeks for geeks, "What is OAuth (Open Authorization) ?," Geeks for Geeks, 11 February 2019. [Online]. Available: https://www.geeksforgeeks.org/what-is-oauth-open-authorization/.

[36] Auth0, "Which OAuth 2.0 Flow Should I Use?," [Online]. Available: https://auth0.com/docs/get-started/authentication-and-authorization-flow/which-oauth-2-0-flow-should-i-use#is-the-client-a-web-app-executing-on-the-server-.

[37] auth0, "Resource Owner Password Flow," [Online]. Available: https://auth0.com/docs/get-started/authentication-and-authorization-flow/resource-owner-password-flow.

[38] Cloudflare, "What is Encryption," Cloudflare, [Online]. Available: https://www.cloudflare.com/learning/ssl/what-is-encryption/.

[39] P. K. DuttaPramanik, G. Pareek and A. Nayyar, "Security and Privacy in Remote Healthcare: Issues, Solutions, and Standards," in *Telemedicine Technologies*, 2019, pp. 201-225.

[40] Z. Wei, "Research on the application of Open source software in Digital Library," *Procedia Engineering*, vol. 15, pp. 1662-1667, 2011.

[41] Q. Jiang., J. Qin. and L. Kang, "A literature review for Open Source Software Studies," *Lecture Notes in Computer Science*, pp. 699-707, 2015.

[42] M. Nelson, R. Sen and C. Subramaniam, "Understanding open source software: A research classification framework," *Communications of the Association for Information Systems*, vol. 17, pp. 266-287, February 2006.

[43] E. N. Hahn, "An Overview of Open-Source Software Licenses and the Value of Open-Source Software to Public Health Initiatives," *JOHNS HOPKINS APL TECHNICAL DIGEST,* vol. 32(4), 2014.

[44] J. Bitzer, W. Schrettl and P. J. H. Schröder, "Intrinsic Motivation in Open Source Software Development," *Journal of Comparative Economics,* vol. 35, no. 1, pp. 160-169, October 2006.

[45] VueJs, "Vue.js - The Progressive JavaScript Framework," [Online]. Available: https://vuejs.org/.

[46] G. Gurung, R. Shah and D. P. Jaiswal, "Software development life cycle models-A comparative study," *International Journal of Scientific Research in Computer Science, Engineering and Information,* pp. 30-37, 2020.

[47] N. S. Yadav, V. Goar and M. Kuri, "AGILE METHODOLOGY -A PERFECT SDLC MODEL WITH SOME IMPROVEMENTS," *Journal of Critical Reviews,* vol. 7, no. 19, pp. 2511-2514, August 2020 .

[48] R. Sherman, "Project Management," *Business Intelligence Guidebook,* pp. 449-492, 2015.

[49] "Manifesto for Agile Software Development," Agile Manifesto, 2001. [Online]. Available: https://agilemanifesto.org/.

[50] P. Adi, "Scrum Method Implementation in a Software Development Project Management," *International Journal of Advanced Computer Science and Applications,* vol. 6, no. 9, September 2015.

[51] S. Sylesh, "A Study of Software Development Life Cycle Process Models," *SSRN Electric Journal,* 10 June 2017.

[52] Lucid Chart, "How to Design Software Architecture," Lucid Chart, [Online]. Available: https://www.lucidchart.com/blog/how-to-design-software-architecture.

[53] Y. Waykar, "A Study of Importance of UML diagrams: With Special Reference to Very Large-sized Projects," in *International Conference on Reinventing Thinking beyond boundaries to Excel*, Faridabad, 2013.

[54] H. Koc, A. M. Erdoğan, Y. Barjakly and S. Peker, "UML Diagrams in Software Engineering Research: A Systematic Literature Review," *Proceedings*, vol. 74, no. 13, March 2021.

[55] M. A. M. Daril, R. Kassim, N. Afiqah S and M. I. A. , "A Development of Logical Design Flowchart for Computerized System of Problem Solving and Improvement Procedure," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 1.1, pp. 400-407, March 2020.

[56] "What is a Flowchart," Lucid, [Online]. Available: https://www.lucidchart.com/pages/what-is-a-flowchart-tutorial.

[57] Ó. S. Ramón, J. S. Cuadrado, J. Vanderdonckt and J. G. Molina, "GUI Generation from Wireframes," September 2013.

[58] P. Martinez, "What is Paper Wireframe and How to Make a Beautiful Paper Wireframe," 25 September 2020. [Online]. Available: https://mockitt.wondershare.com/wireframe/paper-wireframe.html#:~:text=%20How%20to%20Make%20a%20Paper%20Wireframe%20,an%20idea%20of%20the%20number%20of...%20More%20.

[59] G. f. Geeks, "5 Simple Steps to Create Wireframe in Software Design," 22 March 2021. [Online]. Available: https://www.geeksforgeeks.org/5-simple-steps-to-create-wireframe-in-software-design/.

[60] D. I, "What Is a Sitemap & How to Create One for Your Website," 2 February 2022. [Online]. Available: https://www.hostinger.com/tutorials/what-is-a-sitemap.

[61] A2 Hosting, "What Is A Sitemap (And Why Do They Matter)?," 12 September 2018. [Online]. Available: https://www.a2hosting.com/blog/what-are-sitemaps/.

[62] Schlumberger, "Schlumberger," Schlumberger Limited, [Online]. Available: https://www.slb.com/.

[63] P. Pšenák and M. Tibensky, "The usage of Vue JS framework for web application creation," pp. 61-72, January 2020.

[64] B. Behera, "Comparison Between React, Angular and Vue," Medium, [Online]. Available: https://medium.com/pixance-studios/comparison-between-react-angular-and-vue-be488478cbfd.

[65] J. Manhas, "Comparative Study of Cross Browser Compatibility as Design Issue in Various Websites," *BVICAM's International Journal of Information Technology (BIJIT)*, vol. 07, no. 1, pp. 815-820, January 2015.

[66] R. Hartson and P. S. Pyala, "Introduction," in *The UX Book*, 2012, pp. 1-46.

[67] J. M. Caroll, "CHAPTER 1 - Introduction: Toward a Multidisciplinary Science of Human-Computer Interaction," in *HCI Models, Theories, and Frameworks*, 2003, pp. 1-9.

[68] MacKenzie and I. Scott, "Chapter 1 - Historical Context," in *Human-computer Interaction*, 2013, pp. 1-26.

[69] S. R. Venna, R. N. Gottumukkala and V. V. Raghavan, "Chapter 3 - Visual Analytic Decision-Making Environments for Large-Scale Time-Evolving Graphs," in *Handbook of Statistics*, vol. 35, 2016, pp. 81-115.

[70] J. Wang, "From self-efficacy to human-computer interaction design," *Journal of Physics: Conference Series*, vol. 1168, no. 3, February 2019.

[71] B. Shneiderman, "The Eight Golden Rules of Interface Design," University of Maryland, [Online]. Available: https://www.cs.umd.edu/users/ben/goldenrules.html.

[72] S. Herbold, U. Bünting, J. Grabowski and S. Waack, "Deployable Capture/Replay Supported by Internal Messages," *Advances in Computers*, vol. 85, pp. 327-367, 2012.

[73] C. Eaton, "Advances in Web Testing," *Advances in Computer*, vol. 75, pp. 281-306.

[74] F. Lonetti and E. Marchetti, "Emerging Software Testing," *Advances in Computer*, vol. 108, pp. 91-143, 2018.

[75] Vue.js, "Vue.js," Vue.js, [Online]. Available: https://v2.vuejs.org/v2/guide/testing.html.

[76] L. N. Sabaren, M. N. Mascheroni, C. L. Greiner and E. Irrazábal, "A Systematic Literature Review in Cross-browser Testing," *Journal of Computer Science and Technology*, vol. 18, no. 1, 2018.

[77] "Model–view–controller," wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller.

[78] B. Shneiderman, Designing the User Interface Strategies for Effective Human-Computer Interaction, 2016.

[79] E. Wong, "User Interface Design Guidelines: 10 Rules of Thumb," Interaction Design Foundation, [Online]. Available: https://www.interaction-design.org/literature/article/user-interface-design-guidelines-10-rules-of-thumb.

[80] Fusion Charts, "Google Charts vs Chart.js," Fusion Charts, [Online]. Available: https://www.fusioncharts.com/javascript-charting-comparison/google-charts-vs-chartjs.

[81] ChartJs, "Comparison with Other Charting Libraries," ChartJs, [Online]. Available: https://www.chartjs.org/docs/2.9.4/notes/comparison.html.

[82] "Integration Testing and Systems Testing Research Paper," [Online]. Available: https://studentshare.org/statistics/1721737-integration-testing-and-systems-testing-the-roles-than-play.

[83] M. Oberlehner, "Integration Testing Vue.js Components With Jest and Puppeteer," 3 March 2019. [Online]. Available: https://markus.oberlehner.net/blog/integration-testing-vue-components-with-jest-and-puppeteer/.

APPENDICES

Flowchart

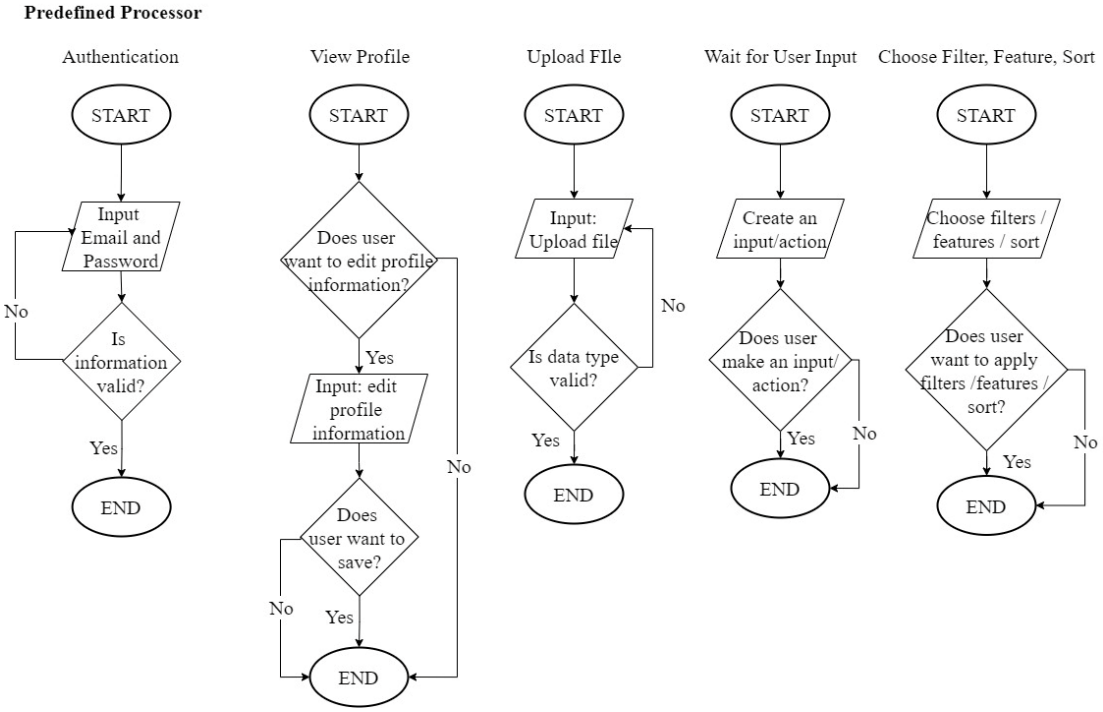


Figure A. 1 Predefined Processors part 1

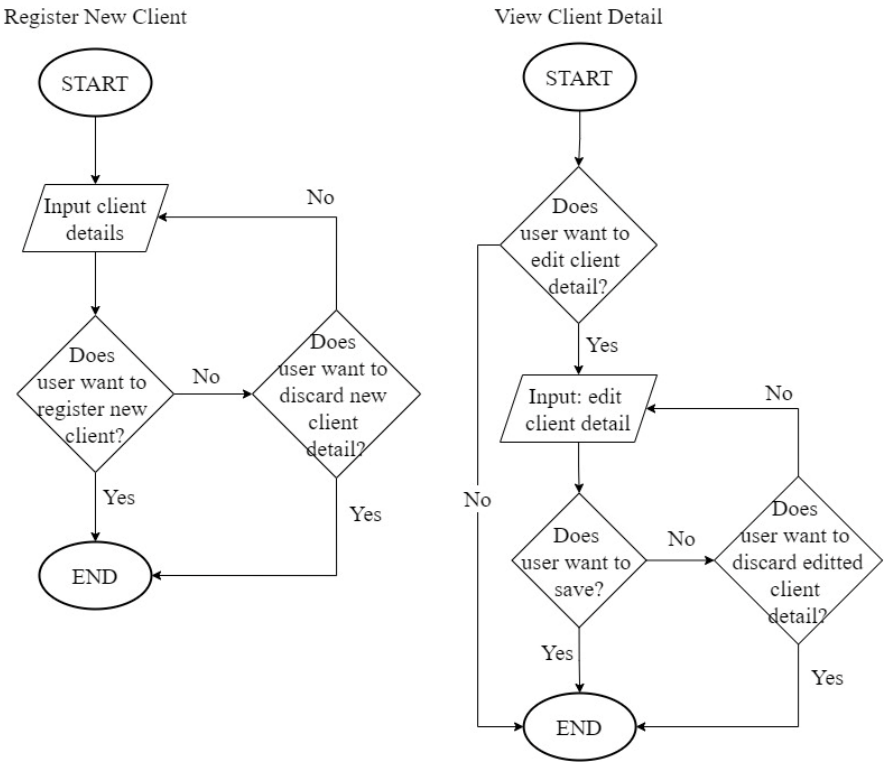


Figure A. 2 Predefined Processors part 2

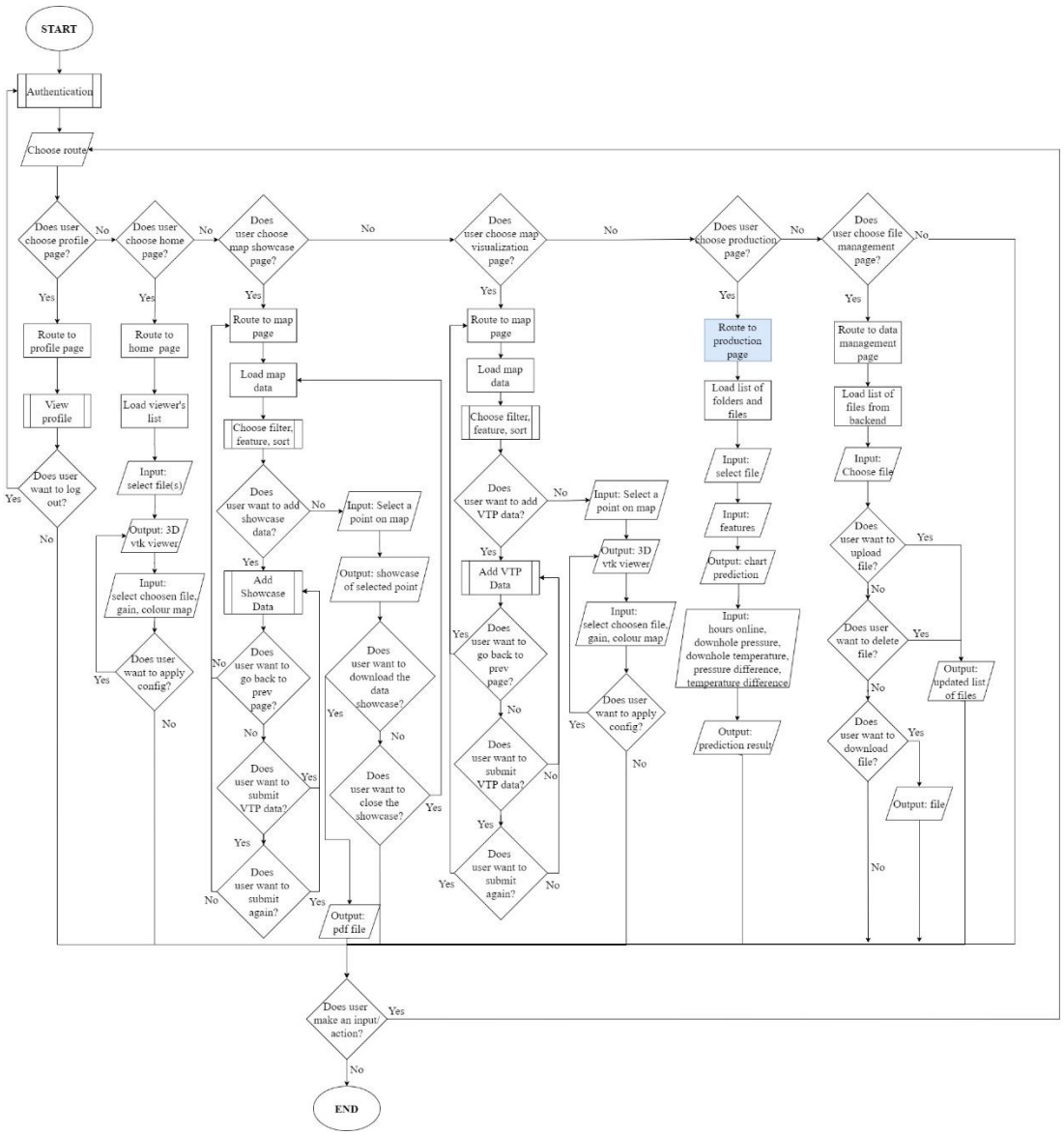


Figure A. 3 Full Flowchart

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

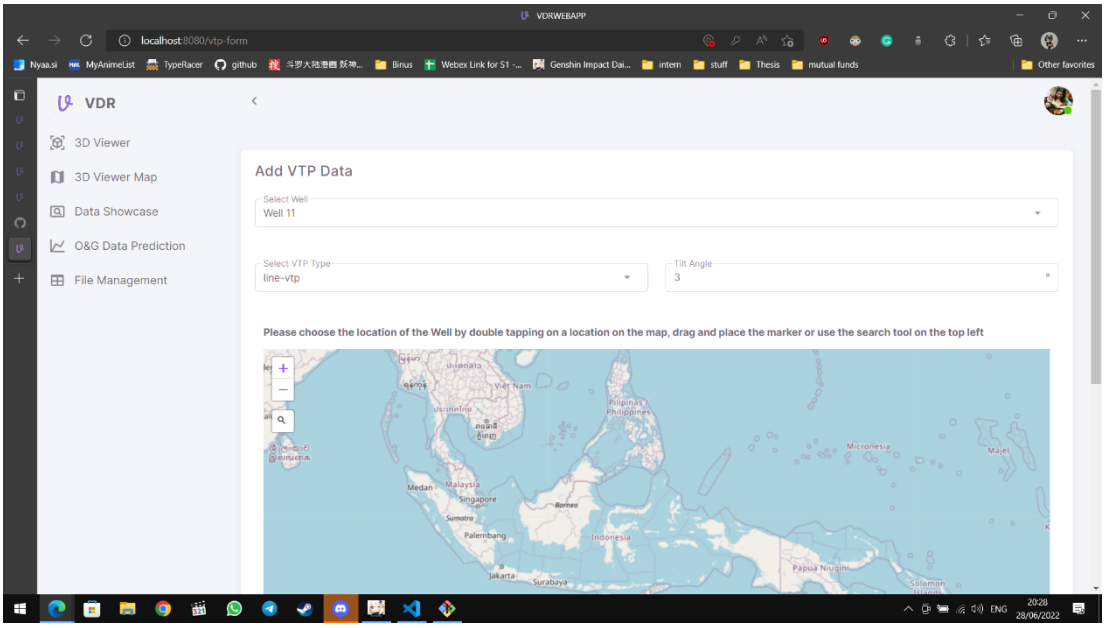
   The user can store their files and store them into folders. These files can and will be used in the other features.

Approved by

Ardimas Andi Purwita	Widodo Nugroho
Product Owner	*Client*

Figure A. 4 Scope of Work Proof

Map Page



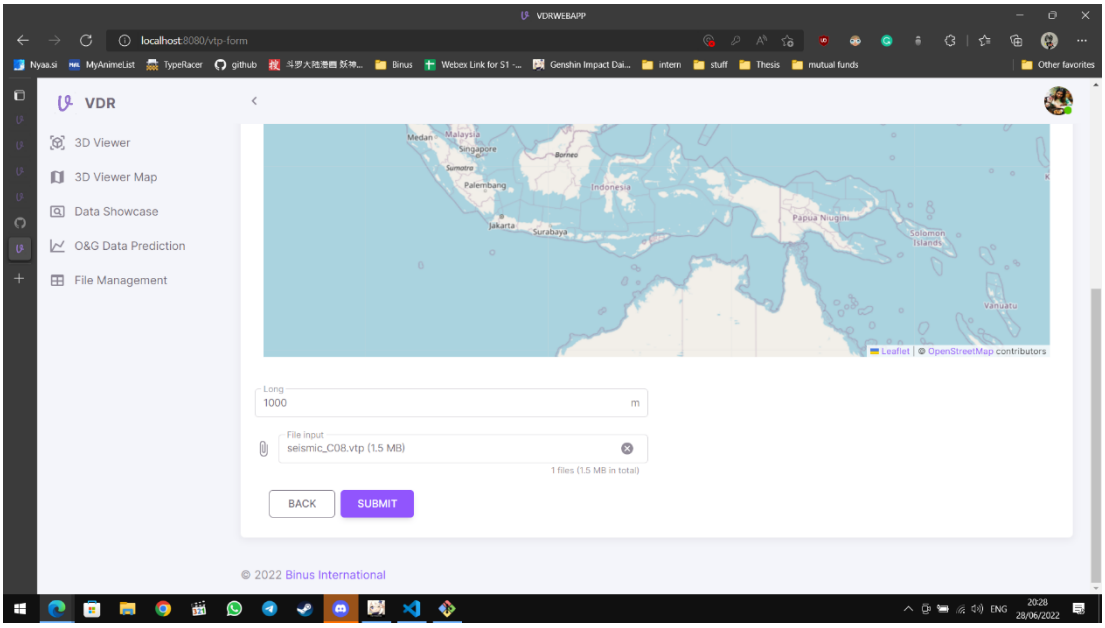


Figure A. 5 VTP Form Page