Black text on a white background

Description automatically generated

**Development of Coal Washability Analysis Application**

**Thesis Report**

Student Name: Jye Doepel

Student Number: 6792984

Subject Code: MMMB498

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

(Materials)

Date of submission:

6th June 2025

# Abstract

Coal washability analysis plays a pivotal role in optimizing coal beneficiation processes, where dense medium separation (DMS) is widely adopted to improve coal quality by removing ash and impurities. This thesis presents the development of a web-based Coal Washability Analysis Application designed to automate the generation and interpretation of coal washability curves from sink-float test datasets. Motivated by the challenges of manual washability analysis, namely its time-consuming nature, susceptibility to human error, and steep learning curve for students and new plant operators, this project provides a digital solution that simplifies calculations, enhances understanding, and supports decision-making in coal preparation.

Built using a modern tech stack comprising React.js for the frontend and FastAPI for the backend, the application integrates scientific libraries such as NumPy and Pandas for efficient mathematical processing, and Plotly for dynamic data visualization. The tool allows users to input raw sink float datasets, visualize densiometric, cumulative floats/sinks, and instantaneous ash curves, and calculate yields and separation densities for multiple coal products. A user centred design approach ensures that the application is both accessible for educational purposes and functional in real-world operational settings. Use cases demonstrate its utility in rapidly determining product yield and quality under various ash content targets. By reducing calculation time, improving accuracy, and making coal washability data interactive and intuitive, this application contributes meaningfully to the digital transformation of the coal industry and supports cleaner, more efficient use of coal resources.

# Acknowledgements

# Declaration

I declare that the work in this thesis project report titled “Development of Coal Washability Analysis Application Thesis Report” has been carried out by me as a part of MMMB498 subject. The information derived from the literature has been duly acknowledged in the text and a list of references provided. No part of this thesis progress report is previously presented for another degree diploma at this or any institution.

Name: Jye Doepel

Student ID: 6792984

University of Wollongong, Australia

Date: 6th July 2025

Table of Contents

[Abstract ii](#_Toc197856168)

[Acknowledgements ii](#_Toc197856169)

[Declaration iii](#_Toc197856170)

[List of Abbreviations vii](#_Toc197856171)

[Chapter 1: Introduction 1](#_Toc197856172)

[1.1. Dense Medium Separation 1](#_Toc197856173)

[1.2. The Coal Washability Curves 2](#_Toc197856174)

[1.2.1. The Specific Gravity Curve (Densiometric Curve) 3](#_Toc197856175)

[1.2.2. The Cumulative Floats Curve 4](#_Toc197856176)

[1.2.3. The Cumulative Sinks Curve 6](#_Toc197856177)

[1.2.4. The Instantaneous Ash Curve (Characteristic Ash Curve) 7](#_Toc197856178)

[1.3. Problem Statement 8](#_Toc197856179)

[1.4. Technical Approach 10](#_Toc197856180)

[1.5. Use Case Scenarios 10](#_Toc197856181)

[Chapter 2: Literature Review 12](#_Toc197856182)

[2.1. Principles of Dense Medium Separation 13](#_Toc197856185)

[2.2.1. Extended Particle Size Range and Capacity 15](#_Toc197856186)

[2.2.2. Improved Cyclone and Circuit Design 15](#_Toc197856187)

[2.2.3. Advanced Medium Density Control and Automation 15](#_Toc197856188)

[2.2.4. Development of Dry DMS Techniques 16](#_Toc197856189)

[2.2.5. Alternative Dense Media and Additives 16](#_Toc197856190)

[2.3. Sink-Float Testing and Washability Curves 17](#_Toc197856191)

[2.4. Advances and Alternatives in Washability Analysis 19](#_Toc197856192)

[2.4.1. Water-Based and Mechanical Methods 20](#_Toc197856193)

[2.4.2. 3D Imaging and Volume–Mass Measurements 20](#_Toc197856194)

[2.4.3. Digital Image Analysis and AI Prediction 21](#_Toc197856195)

[2.4.4. Online Washability Monitoring 21](#_Toc197856196)

[2.5. Digital Transformation in Coal Preparation 22](#_Toc197856197)

[2.6. Software Tools for Data Analysis and Visualization 23](#_Toc197856198)

[2.7. User- Centred Design Considerations 24](#_Toc197856199)

[2.8. Impact of Ash Content on Coal Value and Efficiency 25](#_Toc197856200)

[2.9. Emissions and Environmental Impacts of Ash 27](#_Toc197856201)

[2.10. Conclusion 29](#_Toc197856202)

[Chapter 3: Application Overview 31](#_Toc197856203)

[3.1 Key Feature 31](#_Toc197856204)

[3.2 Purpose and Use Cases 31](#_Toc197856205)

[3.3 User Interface 31](#_Toc197856206)

[3.3.1 Landing Page 32](#_Toc197856207)

[3.3.2 Calculator Page 34](#_Toc197856208)

[3.3.3 About Page 48](#_Toc197856209)

[3.3.4 Reports Page 48](#_Toc197856210)

[3.3.5 Sign In Page 48](#_Toc197856211)

[3.3.6 Sign Up Page 49](#_Toc197856212)

[3.3.7 Settings Page 53](#_Toc197856213)

[Chapter 4: System Architecture and Design 54](#_Toc197856214)

[4.1 System Architecture and Development Cycle 54](#_Toc197856217)

[4.1.1 Development Environment 55](#_Toc197856218)

[4.1.2 Docker Deployment 57](#_Toc197856219)

[4.1.3 Production System 58](#_Toc197856220)

[4.2 Algorithms and Calculations 60](#_Toc197856221)

[4.2.1 Handling Initial Input Coal Sink/Float Data. 61](#_Toc197856222)

[4.2.2 Validating Input Coal Sink/Float Data 64](#_Toc197856223)

[4.2.3 Calculating Information Related to Required Coal Products 65](#_Toc197856224)

[4.2.4 Calculating Information Related to Final Tailing After Producing Coal Products 73](#_Toc197856225)

[4.2.5 Password Cryptography and Authentication of End Users on the Application 74](#_Toc197856226)

[4.3 Technology Stack 76](#_Toc197856227)

[4.3.1 Frontend 76](#_Toc197856228)

[4.3.2 Backend 84](#_Toc197856229)

[4.3.3 Database 85](#_Toc197856230)

[Chapter 5: Conclusion 86](#_Toc197856231)

[References 87](#_Toc197856232)

[Appendices 93](#_Toc197856233)

**No table of figures entries found.**

# List of Abbreviations

|  |  |
| --- | --- |
| **AI** | Artificial Intelligence |
| **API** | Application Programming Interface |
| **AUD** | Australian Dollar |
| **CSS** | Cascading Style Sheets |
| **DMS** | Dense Medium Separation |
| **DMC** | Dense Medium Cyclone |
| **HTML** | Hypertext Markup Language |
| **IP** | Internet Protocol |
| **ISO** | International Organization for Standardization |
| **JSON** | JavaScript Object Notation |
| **JWT** | JSON Web Token |
| **ML** | Machine Learning |
| **RAM** | Random Access Memory |
| **SG** | Specific Gravity |
| **UI** | User Interface |
| **UX** | User Experience |
| **VS Code** | Visual Studio Code |
| **XRF/XRD** | X-ray Fluorescence / X-ray Diffraction |

# Chapter 1: Introduction

## Dense Medium Separation

Coal is a key component in many critical industries, facilitating the operation of much of our essential infrastructure (Minerals Council of Australia, 2024). For this coal to be effective in any critical application, the raw coal extracted from mines must undergo a cleaning process to remove impurities such as ash, sulphur, and moisture (Groppo, 2017). Through the removal of these impurities, the coals energy efficiency, environmental impact, and marketability are all ideally optimized (Groppo, 2017). This process of coal cleaning is known as coal beneficiation or coal washing and one of the most common techniques for this process is the dense medium separation (Groppo, 2017).

DMS utilises the differences in density between the lighter coal and shale (ash), separating out several different fractions (or products) of the raw mined coal, each with a different composition of ash (Meyer & Craig, 2014). The principle of DMS is a simple one, a bath with a density between the desired and undesired material is developed and the raw coal is added. The lighter material (lower ash content) will float, and the heavier material (higher ash content) will sink (Groppo, 2017). In general, it is also necessary for the products of the DMS process to undergo further washing in order to remove and recycle the suspension solids that are undesired for the final product (Groppo, 2017).

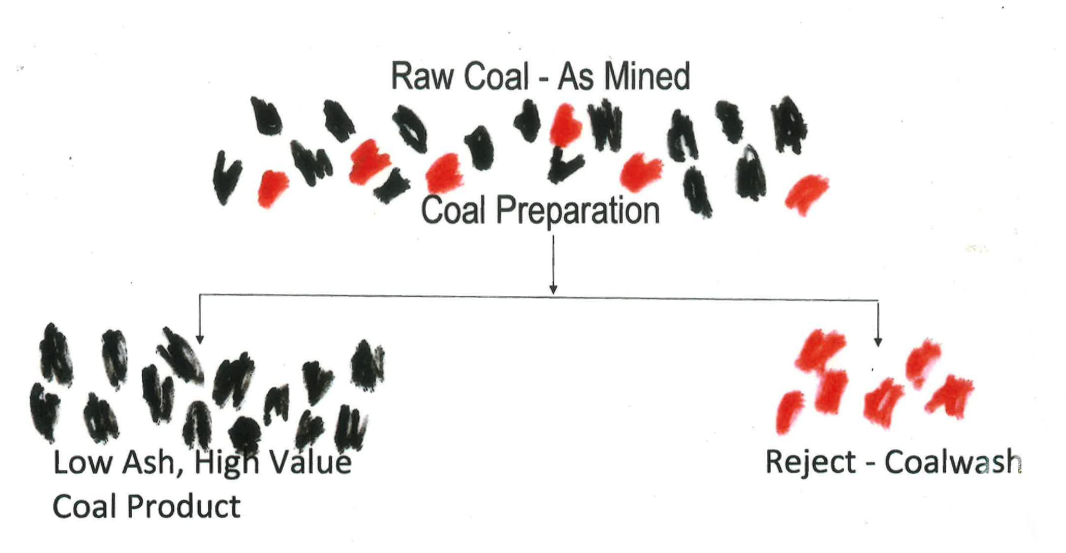


Figure ‎0.1: Illustration of the principle of dense medium separation

The accuracy of the separation in the DMS process is crucial for coal preparation plants in order to optimize both economic value of their products and marketability, as well as desired performance in operation (Burton, et al., 1991). Figure ‎0.2 shows a rough relationship between AUD value of the coal product and the ash content of the product coal. In reality, the exact value of a given coal product is very dynamic and a function of a number of variables (Lin, 2023), not just its ash content, however, this can be used as a rough guide. Figure ‎0.2 illustrates that a reduction in ash content of coal exponentially increase the price of the coal product.

A graph showing the value of a market

Description automatically generated

Figure ‎0.2: AUD$ per tonne of coal fraction against ash content

For a coal preparation plant to decide on its optimized separation density, an analysis of the raw coal input must be undertaken. The analysis typically used for dense medium separation plants is a coal washability analysis (SAHU, 2013). This analysis involves the performance of sink-float tests on a representative sample of the raw coal input across a range of densities. The floats and sinks of each test are analysed to determine their ash percentage. This data can be used to produce a number of washability curves that together can allow coal preparation plant operators to make educated decisions around optimal separation densities (Tolhurst, 2024).

## The Coal Washability Curves

As ash is more dense than clean coal, the ash content of a fraction of coal increases as the density of the fraction increases (Kentucky Geological Survey, 2024). This feature can be exploited through the use of sink float trials (Kentucky Geological Survey, 2024). Small scale tests are carried out using large beakers to determine the coal ash content at different specific gravities, typically ranging from 1.3 - 1.9 SG (Tolhurst, 2024).

Starting at the light or heavy end of the beakers typically depends on where the majority of the material will be removed the earliest (Tolhurst, 2024). As samples need to be dried between, removing as much material as early as possible will reduce the time required for drying. If the lightest end is choses, the coal sample will be first placed in the beaker with the lowest specific gravity. The floats of this trial are scooped off the surface, dried, and analysed for ash content. The sinks are then strained, allowed to dry, and then added to the next beaker of increasing specific gravity. Again, the floats in the beaker are scooped off, dried and analysed and the sinks are poured through a strainer, dried, and placed into the next beaker. This is continued for the remainder of the beakers (Tolhurst, 2024).

From the results obtained from the ash content of each fraction, four washability curves can be developed (Kentucky Geological Survey, 2024):

1. The Specific Gravity Curve (Densiometric Curve)
2. The Cumulative Floats Curve
3. The Cumulative Sinks Curve
4. The Instantaneous Ash Curve

### The Specific Gravity Curve (Densiometric Curve)

The specific gravity curve (Figure ‎0.3) shows the relationship between the density of the separation and the yield of clean coal (floats) (Kentucky Geological Survey, 2024). Thus, if the density of separation is known, then the yield of clean coal can also be found from the graph. By convention, the specific gravity axis is typically on the top horizontal axis and the cumulative weight percentage of floats (yield of clean coal) is typically on the left vertical axis.

A graph with a line going up

Description automatically generated

Figure ‎0.3: 1.2.1. The Specific Gravity Curve (Densimetric Curve)

### The Cumulative Floats Curve

The cumulative floats curve gives the relationship between the yield of clean coal (floats) and the ash content of the coal (Tolhurst, 2024). Thus, if the required ash content is known, then the yield or cumulative weight percentage of floats can be found from the graph. Conversely, if the yield of clean coal is known, then the expected ash content can be found from the graph. By convention, the ash content axis is typically on the bottom horizontal axis and the cumulative weight percentage of floats (yield of clean coal) is typically on the left vertical axis.

A graph with a red line

Description automatically generated

Figure ‎0.4: The Cumulative Floats Curve

The cumulative floats curve can be used with the densiometric (specific gravity) curve. If the ash content of the clean coal yield is known, then from the cumulative floats curve, the yield can be obtained. This value for the yield can then be used to find the density of the separation required from the densiometric curve (Tolhurst, 2024).

A graph with a line going up

Description automatically generated

Figure ‎0.5: The Specific Gravity Curve (Densimetric Curve) & The Cumulative Floats Curve

### The Cumulative Sinks Curve

The cumulative sink curve gives the relationship between the yield of sinks and the ash content of the sinks. So, if the yield of the sinks is known, then the ash content of the sinks can be found from the graph. Conversely, if the ash of the sinks is known then the yield of sinks can be found from the graph.

A graph with a line going up

Description automatically generated

Figure ‎0.6: The Cumulative Sinks Curve

The cumulative sink curve can be used with the cumulative float curve and the densimetric curve. If the ash of the floats is known, then the yield of the floats can be obtained from the cumulative floats curve. If the density is known, then the yield of floats can be obtained from the densiometric curve. In either case the yield of sinks is known, so the ash of the sinks can be found from the cumulative sinks curve.

A graph with different colored lines

Description automatically generated

Figure ‎0.7: The Specific Gravity Curve (Densimetric Curve), The Cumulative Floats Curve & The Cumulative Sinks Curve

### The Instantaneous Ash Curve (Characteristic Ash Curve)

The Instantaneous ash curve (characteristic ash curve) gives the relationship between the yield of the floats and the ash content of the particle that just floats or just sinks at that yield. It thus gives the highest ash in the floats or the lowest ash in the sinks.

A graph with a line going up

Description automatically generated

Figure ‎0.8: The Cumulative Sinks Curve

The instantaneous ash curve can be used with the other three washability curves, the specific gravity curve, the cumulative float curve, and the cumulative sink curve.

A graph with different colored lines

Description automatically generated

Figure ‎0.9: The Combined Coal Washability Curves

## Problem Statement

Accurate interpretation of these washability curves is essential for coal preparations operators to determine the optimal separation densities for their desired coal products. Issues can however arise when through manual interpretation of the raw coal washability analysis data as interpretation involves manually complex mathematical calculations, which can be time-consuming and prone to error. It is here that the use of automated computer application for the analysis of this coal washability data is crucial in optimising both the quality and economic value of the final products.

The complexity of manual calculations poses several challenges:

1. **Time and Labor**: Processing large datasets of coal properties is a slow and tedious process that takes up valuable time for plant operators and engineers.
2. **Error Potential**: The manual nature of these calculations increases the likelihood of mistakes, leading to inaccurate washability curves.
3. **Educational Barriers:** For students or newcomers to coal preparation processes, understanding how to conduct and interpret washability analysis can be overwhelming without automated tools.
4. **Decision-Making Delays:** The time-consuming nature of manual calculations can delay key decision-making processes in coal preparation plants, potentially leading to inefficiencies and lower product yields

Given these challenges, there is a clear need for a modern, digital tool that can automate coal washability analysis, making it more accessible and efficient for both students and industry professionals. This project addresses that need by developing a computer-based computational model for generating coal washability curves, which eliminates the need for manual calculations.

The main goal of this project is to develop an easy-to-use application that automates coal washability calculations and provides interactive visual representations of coal washability curves. The application will serve as a resource for both students learning about coal beneficiation processes and plant operators who need to quickly and accurately determine the best strategies for coal separation.

At its core, the project seeks to address the following objectives:

1. **Automation of Washability Analysis**: The application will automate the process of generating coal washability curves based on user-provided data, such as ash content and weight percentages at various specific gravities.
2. **Interactive Visualization:** The tool will provide an intuitive, interactive interface that allows users to input data and view the resulting coal washability curves in real-time. This will enable users to understand how different separation strategies impact coal quality and yield.
3. **Reduction of Manual Calculations:** By automating the washability analysis, the tool will eliminate the need for manual mathematical calculations, saving users time and reducing the risk of errors.
4. **Enhanced Educational Resource:** For students, the application will serve as a practical learning tool, helping them understand coal preparation processes and the significance of washability analysis in industrial applications.
5. **Support for Decision-Making:** For plant operators, the application will provide a fast and reliable way to determine optimal separation strategies, aiding in decision-making and improving plant efficiency.

## Technical Approach

The proposed application will blend modern web-based technologies and data processing tools to ensure both a user-friendly product and a product that can handle the complex calculations. On the user facing (front end) side, the popular JavaScript framework React.js will be employed. React.js simplifies building dynamic user interface by using a component-based architecture (Gackenheimer, 2015). This decision will assist in management of the user input forms, washability plot visualizations, and streamlined communication with the backend system. Users will have the ability to upload a set of raw sink-float data in a range of formats and enter in the ash percentages of the coal products they desire. The integration of the JavaScript library Plotly will allow for the visualization of the washability curves in a dynamic and interactive format (Li & Bilal, 2021). This integration allows users to delve into the graphs and adjust their inputs to observe how different values impact the results.

For the backend system, the modern python-based web framework FastAPI is to be used as it is ideal for building high-performance APIs. The backend system’s primary role is to receive requests from the frontend React.js system (the raw sink-float data), provide some data processing, and return the data to be plotted. The combination of FastAPI as a backend system combined with React.js for a frontend system is quickly becoming a popular technology stack for building quick and easy web applications that are also robust and have a modern feel (Gailer, et al., 2024).

For the complex mathematical calculations and interpretations of the raw sink-float washability data, the python libraries Numpy and SciPy will be utilized. NumPy excels at efficiently handling numerical data and large arrays, while SciPy offers tools for tasks such as curve fitting and optimization (Ranjani, et al., 2019).

## Use Case Scenarios

To illustrate the practical application of the tool, the following scenario should be considered: A customer requires a Coking Coal with 9.0% ash. The coal plant operator needs to determine the yield of clean coal and the specific gravity at which the separation should occur. Using the application, the operator can input the relevant ash percentages and raw sink-float data obtained by laboratory tests, and the tool will instantly generate a coal washability curve showing the yield of clean coal and the required density of separation.

If a secondary Steaming Coal product with a 25.0% average ash can be sold, the operator can use the application to add in the second coal product and calculate the specific gravity for this separation and estimate the percentage of feed that will be recovered as Steaming Coal. The application will also provide the remaining ash percentage in the tailings, allowing the operator to evaluate the efficiency of the entire coal preparation process.

# Chapter 2: Literature Review

Coal remains a critical energy resource globally, used extensively for electricity generation, steelmaking (as coke), and industrial heating​ (Phengsaart, 2022). However, raw coal is typically mined with significant impurities – rock fragments, clays, pyrite, and other minerals – that form ash upon combustion and contribute to pollution (e.g. sulphur oxides from pyrite, additional CO₂ from heating inert material (Phengsaart, 2022). To improve the quality of coal and mitigate environmental impacts, coal beneficiation (or coal cleaning) is employed as a pre-combustion strategy to remove these impurities. Among coal beneficiation techniques, gravity separation methods that exploit density differences are widely practiced due to their simplicity, cost-effectiveness, and high efficiency (Phengsaart, 2022). In particular, Dense Medium Separation(DMS) – separating coal and waste in a dense fluid medium – is one of the most popular and effective processes in modern coal preparation plants​ (Phengsaart, 2022).

A thorough understanding of coal “washability” is foundational for designing and optimizing such separation processes. *Washability analysis* refers to the evaluation of a coal sample’s response to density-based separation, typically via float-and-sink testing that reveals how much coal can be recovered at various density cut-points and what product quality (ash content) can be achieved​ (Kentucky, 2025) (Galvin, 2025). The results are usually presented as washability curves, which serve as a benchmark for the best possible separation performance achievable for that coal (Galvin, 2025). With the advent of digital transformation in the mining industry, there is also a growing trend to integrate software tools for data analysis, visualization, and even real-time monitoring of coal quality. Modern computational techniques, including digital twin simulations and image analysis, are being explored to enhance or even replace traditional laboratory methods (Lin, et al., 1998) (Botlhoko, et al., 2022) ​. Moreover, any development of a new coal washability analysis application must consider not only the technical principles of separation and data presentation, but also user-centered design for usability and the broader economic and environmental context in which coal beneficiation operates.

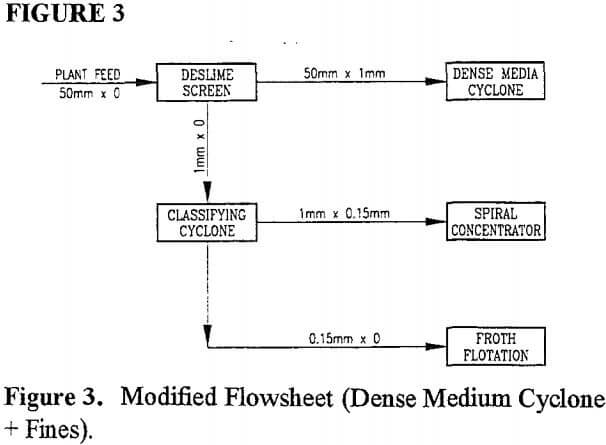
In this literature review, we survey the key principles and recent advancements in Dense Medium Separation as applied to coal; examine methodologies for coal washability analysis (with emphasis on float–sink tests and washability curve interpretation); explore the integration of software and digital tools in coal analysis (including trends in data visualization and user-focused design); and discuss the economic and environmental considerations of coal beneficiation, particularly how ash content influences coal value, energy efficiency, and emissions. This review draws on recent peer-reviewed studies, industry reports, and technical papers to provide a comprehensive overview appropriate for an undergraduate engineering thesis on developing a coal washability analysis application.



## Principles of Dense Medium Separation

Dense Medium Separation (DMS) is a gravity-based separation technique that separates coal from high-density impurities by immersion in a fluid medium of intermediate density. The principle is straightforward: coal particles, being less dense (specific gravity around 1.2–1.5), tend to float in a suspension if the medium’s density is set between the density of coal and that of the unwanted mineral matter; conversely, the denser rock and pyritic particles (specific gravity often 2.4–5.0) sink (Kentucky, 2025). In practical applications, the dense medium is usually a finely ground heavy mineral (most often magnetite, Fe₃O₄, or sometimes ferrosilicon) mixed with water to form a slurry of controlled specific gravity, typically in the range 1.3–1.8 g/cm³. When raw coal is introduced into this suspension, the low-ash coal “floats” to the top and is recovered as product, while the high-ash material “sinks” to the bottom and is rejected (Kentucky, 2025). The float and sink products can be separately collected, dewatered, and analysed for yield and ash content, mirroring on a continuous scale what a float–sink test does in discrete density steps.

In coal preparation plants, DMS is implemented in specialized vessels or circuits. Two common types of dense medium separators are employed based on particle size range: dense medium drums or baths for coarse coal, and dense medium cyclones (DMC) for intermediate sizes. In a typical plant flowsheet, the run-of-mine coal is screened into size fractions, for example: a coarse fraction (e.g. >50 mm) processed in a heavy medium bath or drum, an intermediate fraction (e.g. 50–0.5 mm) treated in dense medium cyclones, and fine or ultrafine fractions handled by other methods (like water-based spirals, teeter-bed separators down to ~0.15 mm, and froth flotation for ultrafine <0.15 mm) (David, 2019). The dense medium cyclone, in particular, has become the workhorse for coal cleaning, as it can rapidly separate particles within a wide size range with high precision. Coal slurry (mixed with magnetite medium) is fed tangentially into the cyclone, creating a vortex that subjects particles to centrifugal forces; lighter coal moves toward the centre (axial flow, reporting to the overflow or “float” stream), while heavier refuse is flung outward to the cyclone wall and exits via the underflow (or “sink” stream). By adjusting the medium density, the operator can control the effective cut-point (separation density) to target a desired coal product quality.



#### Modified Flowsheet (Dense Medium Cyclone + Fines) (David, 2019)

The popularity of DMS in coal beneficiation stems from its efficiency and scalability. DMS can achieve sharp separations of coal from high-ash impurities even at high throughput, and it is particularly effective for difficult-to-wash coals that may have a narrow range of density differences. Studies show that among traditional coal cleaning technologies, dense medium separators (especially DMCs) have been the most researched and implemented in the last decade, which is attributable to the need for cleaning increasingly fine and challenging coal feeds (Phengsaart, 2022). In essence, a well-operated dense medium circuit can recover a large proportion of combustibles while rejecting most mineral matter, coming close to the ideal washability performance of the coal. Modern coal preparation plants often rely on DMS as the primary stage of beneficiation, given its reliability and the strong theoretical foundation in density separation.

* 1. **Recent Advancements in DMS Technology for Coal**

Over the years, significant advancements have been made to improve the performance, range, and cost-effectiveness of dense medium separation in coal preparation. These advancements span equipment design, operating techniques, control systems, and even alternative “dense” media. Key recent developments include:

### Extended Particle Size Range and Capacity

Innovations in DMS equipment have pushed the envelope of the particle sizes that can be economically treated. Traditionally, dense medium cyclones were limited to a top size of ~50 mm, but since the 1990s, manufacturers have introduced larger-diameter cyclones (>900 mm) capable of handling coal up to 75–100 mm in size (David, 2019). This, combined with more efficient pre-screening and feed preparation, has enabled DMS to treat a broader size spectrum from coarse lumps down to fine coal (~0.15 mm) within a single process​ (David, 2019). The increase in cyclone diameter and improved volumetric capacity allow much higher throughput. For instance, replacing a 710 mm cyclone with a 1000 mm unit can dramatically increase the feed rate, as shown by case studies in large coal plants (David, 2019). Extending DMS to a finer top size also reduces the burden on auxiliary circuits (like jigs, spirals, or flotation), simplifying the overall plant flowsheet.

### Improved Cyclone and Circuit Design

A deeper understanding of how cyclone geometry and ancillary equipment affect performance has led to design optimizations. Research on cyclone configuration (e.g. cone angle, vortex finder and spigot dimensions, etc.) and the ratio of cyclone components has helped minimize misplacement of particles and improve the sharpness of separation​ (David, 2019). Additionally, improvements in the design and capacity of desliming screens (used to remove ultra-fines before DMS) and drain-and-rinse screens (which recover magnetite from product and reject streams) have enhanced circuit efficiency. Modern high-frequency screens can effectively remove clay-rich slimes that would otherwise contaminate the medium, and they can operate at higher capacity without blinding (David, 2019). Better magnetite recovery systems – such as more powerful wet drum magnetic separators – reduce magnetite losses and keep the medium density stable, lowering operating costs (David, 2019). All these hardware improvements make dense medium circuits more robust and cost-efficient than in decades past.

### Advanced Medium Density Control and Automation

Because the performance of a DMS circuit is highly sensitive to the medium’s specific gravity, precise control of medium density is critical. Recent advancements include automated density control systems and algorithms that adjust medium composition in real time to compensate for feed variability (David, 2019). For example, modern plants may use nuclear density gauges or digital pressure transmitters on cyclone feed and overflow to infer the medium density, automatically adding water or magnetite to keep the density on setpoint. In the research domain, sophisticated control approaches have been proposed, such as model-predictive or adaptive controllers that account for the nonlinear and time-varying nature of DMS (Dai, 2020). One study introduced a model-and-data switching adaptive control to continuously tune the medium density set-point for producing low-ash coal, demonstrating improved stability and responsiveness in a simulation with real plant data (Dai, 2020). Such automation ensures the DMS operates at optimal cut-point and efficiency even as feed coal characteristics change, thereby maximizing yield of clean coal.

### Development of Dry DMS Techniques

A notable area of innovation is water-less dense medium separation for regions where water is scarce or where a wet process is impractical. Conventional DMS requires large volumes of water, which can be a limiting factor in arid mining regions or where tailings water disposal is problematic (Hughes, et al., 2024) . In response, researchers have developed *Air Dense Medium Fluidized Beds (ADMFB)* and other dry separation methods. In an ADMFB, a solid medium (e.g. magnetic or sand particles) is fluidized by air to create a pseudo-fluid of controllable density. Coal introduced into this fluidized bed will stratify by density similar to a wet process. Studies from China and elsewhere have shown that an air-fluidized medium bed can efficiently beneficiate coal in the 50–6 mm size range, with separation densities adjustable between about 1.2 and 2.2 g/cm³ (Wei, 1998)​. The efficiency of separation in some dry systems is quite competitive; for example, a probable error (*Eₚ* ≈ 0.06) has been reported in pilot dry separators, indicating sharp separations ​ (Wei, 1998). Dry DMS eliminates the need for process water and the associated dewatering of products, making it attractive for remote or drought-prone areas. A recent systematic review (Hughes, et al., 2024) concluded that several dry coal-beneficiation methods (from pneumatic tables to optical sorting and ADMFBs) are nearing commercial viability, and with further refinement they could be deployed in future coal preparation plants in cold, arid, or developing regions.

### Alternative Dense Media and Additives

While magnetite is the industry standard medium, there have been experiments with other materials to address cost or performance issues. One novel advance is the use of soluble heavy salts to create a dense liquid medium. Calcium nitrate (a common, inexpensive fertilizer) was recently demonstrated as a dense medium for low-grade coal upgrading in Colombia (Daley, et al., 2022). In that study, lumps of very high-ash coal (up to 29% ash) were immersed in concentrated calcium nitrate solution; the float product’s ash was reduced to below 7%, a remarkable improvement sufficient to meet metallurgical coal specifications (Daley, et al., 2022). Calcium nitrate solutions can be made very dense (>1.5 g/cm³) and offer a low-cost alternative where magnetite might be expensive or unavailable ​ (Daley, et al., 2022). There are challenges, of course, including recovery and recycling of the dissolved salt and potential corrosion or toxicity issues, but research is ongoing to mitigate these (e.g. exploring recovery of the salt or using other benign high-density salts) (Daley, et al., 2022). Another area is the use of stabilizing additives in the medium to improve separation of ultra-fine coal. For instance, adding surfactants or ultra-fine clay to the medium can help maintain stable suspension and prevent misplacement of fine coal, although such practices are still mostly experimental.

Overall, these advancements have reinforced DMS as a cornerstone of coal beneficiation. Modern DMS circuits are more versatile – able to clean a broader range of coal sizes and qualities – and more precise, thanks to better engineering and control. The focus on dry separation options and novel media also shows the drive to adapt coal cleaning to new constraints (like water scarcity and environmental safety). For an engineer developing a coal washability analysis application, it is important to incorporate these state-of-the-art developments. Such an application might, for example, allow users to input data for different separation techniques (wet or dry) or simulate the effect of tighter control on separation efficiency. A well-rounded understanding of DMS ensures that the software can model or consider real-world plant scenarios, including the capabilities of the latest separation technologies.

## Sink-Float Testing and Washability Curves

The classic method for determining coal washability is the float–sink test (also called sink–float analysis). This laboratory procedure reveals how coal splits into clean coal and refuse across a range of separation densities. In a standard float–sink analysis (as per ASTM D4371 or similar standards), a representative coal sample is first crushed to the particle size at which beneficiation would be performed (e.g. often 50 mm top size for coarse washability, or finer for specific assessments)​ (Kentucky, 2025). The sample is then subdivided and immersed in a series of liquids (or heavy liquid solutions) of known specific gravities, typically incrementing from around 1.30 up to 2.0. Common heavy liquids include organic compounds like tetrachloroethylene (specific gravity ~1.62) or methylene bromide (~2.0), often diluted with lighter solvents to achieve intermediate densities. At each density step, particles with density lower than the liquid will float and those heavier will sink. The floating portion is skimmed off, washed, dried, and weighed to record the float yield at that density; its ash content (and sometimes sulphur or other qualities) is analysed. The remaining sink fraction is then introduced to the next heavier liquid and the process repeats, thus partitioning the sample into multiple density fractions (Kentucky, 2025). Ultimately, data is obtained for the cumulative float (i.e. total coal that would float at or below each density) and the cumulative sink (coal that would sink) with corresponding quality metrics (Kentucky, 2025).

From the float–sink data, various washability curves are constructed to visualize the cleaning potential of the coal. The most fundamental is the cumulative yield vs. cumulative ash curve (Galvin, 2025). This curve plots, for each density cut, the total weight percentage of coal that would be recovered (yield) against the average ash content of that recovered coal. It effectively shows the best possible yield one could achieve for any given product ash. For example, it might show that to obtain a product with 10% ash, at most 80% yield is attainable; trying to recover beyond 80% of the coal would result in product ash above 10%. This yield-ash relationship is often referred to as the cumulative floats curve, and it provides a benchmark for evaluating the performance of actual separation units​ (Galvin, 2025). If an operating dense medium cyclone produces 75% yield at 10% ash, as the cumulative floats curve indicates 80% was theoretically possible, then the separation efficiency is reasonably high (the losses are close to the inherent constraint of the coal). On the other hand, a large gap would signal inefficiency or poor separation (perhaps due to equipment issues or presence of “near-density” material, discussed below).

Another useful plot derived from washability data include: the specific gravity curve or densiometric curve (showing the relationship between the density of the separation and the yield of clean coal (floats)). Thus, if the required ash content is known, then the yield or cumulative weight percentage of floats can be found from the graph. Conversely, if the yield of clean coal is known, then the expected ash content can be found from the graph. By convention, the ash content axis is typically on the bottom horizontal axis and the cumulative weight percentage of floats (yield of clean coal) is typically on the left vertical axis. The cumulative floats curve can be used with the densiometric (specific gravity) curve. If the ash content of the clean coal yield is known, then from the cumulative floats curve, the yield can be obtained. This value for the yield can then be used to find the density of the separation required from the densiometric curve.

A critical concept in interpreting washability data is the presence of near-gravity material (NGM). Near-gravity material is typically defined as the fraction of the coal that has a density very close to the intended separation density (commonly within ±0.1 specific gravity units)​ (David, 2017). This portion of the material is problematic because it is difficult to predict whether it will report to floats or sinks in an industrial separator – small process fluctuations can send it either way. When a large percentage of the feed lies in this “gray area” around the cut-point, achieving a clean separation becomes inherently challenging (David, 2017). Graphically, one can visualize NGM by looking at the density distribution curve: if there is a high peak near the cut density, the separation will be difficult. Quantitatively, an NGM index can be calculated (percentage of feed within ±0.1 SG of cut). Values above, say, 10% NGM indicate a moderately to extremely difficult coal to wash (Bhagwat, 2009). This is why some coals, despite beneficiation, cannot achieve very low ash yields without incurring big losses – their washability curve might drop sharply, reflecting a lot of material clustered near the cut-point density.

From the washability curves, one can determine the optimum separationdensity for a desired outcome. Typically, the goal is to maximize yield for a given product quality constraint (ash or sulphur). The float–sink test is indeed commonly used to identify this optimum cut density (Cheng, et al., 2018). For instance, if a power plant requires coal of 15% ash, the washability data might show that separating at a relative density of 1.6 gives the best yield of such coal. It also allows modelling of multi-product scenarios (e.g. making a middling product). In summary, float–sink analysis and the resultant curves are invaluable for coal preparation engineers: they set the theoretical limits and guide the choice of operating parameters for separators.

## Advances and Alternatives in Washability Analysis

While float–sink testing has been the gold standard for decades, it has some downsides: it is labour-intensive, time-consuming, and uses toxic heavy liquids (like tetra-bromoethane or zinc chloride solutions) which pose health and disposal hazards (Botlhoko, et al., 2022). Furthermore, regulations (such as in Australia) are moving away from the use of hazardous organic liquids for routine analysis (Galvin, 2025). This has prompted research into alternative methods to obtain washability information that are faster, safer, or more automated.

### Water-Based and Mechanical Methods

One approach is to replace toxic heavy liquids with water-based separation. For example, float–sink in dense salt water or magnetized ferrofluid in water have been tested (Galvin, 2025)​. In magnetized ferrofluid methods, a colloidal suspension of magnetic nanoparticles (in water or kerosene) is used; by applying a magnetic field, the effective density of the fluid can be “tuned”, and particles can be levitated at different equilibrium points based on density. Another concept is fluidized bed separation in the lab. A column of water is used to create a density gradient by upward flow (perhaps with fine silica to simulate a teeter bed), and coal sample is introduced to let lighter particles rise and heavier fall. The Australian Coal Association research noted that water-based systems can work but may cause some breakage of soft coal or clay dispersion, altering the washability slightly (Galvin, 2025). However, this arguably simulates what happens in a real wet prep plant, so with careful calibration (like performing a wet size split to account for fine generation), the method can be viable (Galvin, 2025)​. The conclusion of a comprehensive Australian review by Galvin was that satisfactory alternatives to heavy-liquid float–sink do exist – particularly involving water-based apparatus – though each comes with compromises (either in complexity, precision, or cost) (Galvin, 2025). The industry is actively considering these as the likely path forward once traditional float–sink standards are phased out.

### 3D Imaging and Volume–Mass Measurements

A breakthrough in recent years is the application of imaging technology to determine particle density without float–sink. One such method is the RhoVol 3D image-based analyser (Botlhoko, et al., 2022). The principle of RhoVol is to measure the volume and mass of each coal particle directly: high-resolution 3D scans (or multiple camera images) provide the volume, and a sensitive balance gives the mass, yielding density by calculation (Botlhoko, et al., 2022). By doing this for many individual particles from a sample, one can essentially reconstruct the density distribution and thus simulate a float–sink analysis without actually floating the particles in liquids. Botlhoko *et al.* demonstrated this technique on South African coals and found that the washability curves obtained via RhoVol matched closely with those from traditional float–sink tests​ (Botlhoko, et al., 2022). The method proved to be rapid and free of hazardous substances, indicating a promising alternative for routine analysis​ (Botlhoko, et al., 2022). Each particle’s ash content can also be predicted if a correlation with density is known or by additional sensors (like X-ray or optical measurement of composition). The success of RhoVol suggests that future coal labs might use optical or X-ray scanning devices to get washability data in hours instead of days, and without carcinogenic chemicals.

### Digital Image Analysis and AI Prediction

Related to 3D imaging are methods that use surface image analysis of coal particles to infer composition. Research in the 2010s explored taking photographs of crushed coal and using image processing algorithms to classify particles as likely “coal” or “rock” based on colour/texture, then using machine learning (e.g. a radial-basis function network) to predict the washability curve from those features (Ze-lin, et al., 2011). Essentially, if the proportion of dark, bright coal vs. dull, clay-coated rock can be quantified by image analysis, one can estimate how much coal would float at a certain density. One study combined image processing with a neural network and reported reasonable predictions of the washability curves without performing a physical float–sink test (Ze-lin, et al., 2011). Although these methods may not yet be as precise as direct measurement, they illustrate the move toward fast, digital predictive analytics in coal quality assessment.

### Online Washability Monitoring

Beyond laboratory analysis, there have been attempts to measure washability continuously on coal streams. A notable example is research into using X-ray computed tomography (CT) for online washability analysis (Lin, et al., 1998). In this concept, a sample of coal (or the full stream on a conveyor) is scanned by X-rays to determine the distribution of material densities. In the late 1990s, University of Utah and collaborators built a prototype where sequential CT scans of a falling stream of particles were processed by custom software to yield washability data (Lin, et al., 1998)​. The goal was a device that could be installed at a preparation plant to instantly report the float–sink curve of the feed or product, enabling dynamic process control and coal blending decisions (Lin, et al., 1998). While technical challenges (e.g. X-ray resolution, processing speed) meant such a system did not immediately commercialize, the continued improvements in sensor technology suggest it may be revisited. In general, digital transformation in mining is driving the integration of analytical instruments with software – for example, rapid XRF/XRD analysers for mineral content, laser-induced breakdown spectroscopy for coal quality, and on-belt ash gauges. An on-line washability analyser, if fully realized, would be a prime example of this trend, providing real-time data to operators and potentially feeding into an expert system or plant optimizer.

In summary, coal washability analysis is evolving from a purely manual, chemical-based test to a more high-tech domain involving imaging, automation, and software modelling. Any new software application intended for washability analysis should be cognizant of these changes. It could, for instance, incorporate modules for handling data from novel sources like imaging devices or allow for the input of parameters from non-standard tests. The traditional float–sink results will remain highly relevant (as they define the “ground truth” for washability), but the means of obtaining and working with those results are broadening. Consequently, the literature points toward a future where coal washability is determined faster and more safely, and where the interpretation of washability is aided by visualization tools and possibly AI, rather than just static plots on paper.

## Digital Transformation in Coal Preparation

The mining and mineral processing industry, including coal preparation, has been undergoing a digital transformation in recent years. Broadly, this means adopting technologies like advanced sensors, process automation, data analytics, and computer modelling to improve efficiency and decision-making. In coal preparation plants, digital transformation is evident in the use of modern control systems (SCADA and PLC-based controls) that regulate processes like dense medium separation with minimal human intervention. One cutting-edge concept is the creation of a digital twin of a coal preparation plant. A digital twin is a virtual replica of the physical process, updated in real-time with data from the plant, which can be used for simulation, monitoring, and optimization. Researchers have proposed digital twin frameworks for coal preparation that integrate 3D visualization of the plant with real process data and even machine learning models (Zhang, et al., 2021). For example, Zhang *et al.* (2021) describe a digital twin system combining a real-time database of a coal plant with a 3D model; it enabled interactive integration of the physical and informational aspects of the plant and was applied to equipment condition monitoring (like predicting the health of a coal shearer using deep learning on sensor data) (Zhang, et al., 2021). This demonstrates how linking the “information world” with the physical operations can help anticipate maintenance needs and optimize performance in a coal facility.

In the specific context of coal washability analysis and coal quality, digital transformation yields several advantages. Plant operators can now receive continuous feedback on product quality from online analysers (such as nuclear ash gauges, moisture sensors, and elemental analysers) and adjust the plant in real time. Some modern plants use “expert systems” – essentially software that uses rules or AI to suggest optimal settings – to achieve target product specifications. A washability analysis application could tie into such systems by, say, predicting how changes in cut density or feed composition would shift the product quality, thereby serving as a decision support tool. Furthermore, big data techniques can be applied to historical float–sink data and plant results to find patterns or develop empirical models for performance. Overall, the trend is toward intelligent coal preparation: using digital technology to make coal processing more adaptive and efficient.

## Software Tools for Data Analysis and Visualization

A critical aspect of modernizing coal analysis is improving data visualization and accessibility. Traditional washability analyses were often done on paper charts or spreadsheets, but today’s engineers benefit from interactive software that can manipulate data and display results instantly. For instance, some mine planning and geology software suites have incorporated coal washability functionality. Maptek’s Vulcan (a widely used mine planning software) includes a Coal Washability module that allows engineers to integrate washability data (float–sink results) with the geological model of a coal deposit (Maptek, 2020). This means one can query within the geological model what the expected yield and quality would be at various cut points, which in turn aids in resource evaluation and scheduling. The software helps users “better understand their coal quality, improving downstream processes such as scheduling” by providing a detailed overview of washability characteristics and how they vary spatially (Maptek, 2020). Similarly, tools like Micromine’s Geobank database are designed to store and handle large sets of coal quality and washability data for easy retrieval and analysis (Micromine, 2018). These industry tools underscore the importance of integrating washability analysis into a broader data environment – rather than treating it as an isolated lab result, it becomes part of the digital mine model.

Data visualization in these tools often includes generating washability curves on-screen, with options to overlay multiple samples, zoom into specific gravity ranges, or instantly see the impact of choosing a different cut density. The ability to quickly plot cumulative yield vs ash or density distribution histograms helps engineers and stakeholders grasp the cleaning characteristics of a coal seam. Moreover, with computational power, one can simulate the outcome of multi-stage washing. For example, an application might take the float–sink data and simulate a two-stage separation (rougher and cleaner) or produce Tromp curves (which are actual partition curves of separators) and compare them to the ideal washability curve. An academic study from 2023 even demonstrated plotting coal washability curves using Python programming, highlighting that relatively simple scripts can automate the generation of these curves and extraction of key metrics (Cuiping, et al., 2023). This points to the value of open-source tools and programming in customizing analysis – something very relevant for a thesis project aiming to develop a new application.

Beyond static data, visualization can extend to 3D and immersive environments. There are examples of 3D web-based visualization of coal prep plants where users can see a color-coded representation of density or ash content distribution in raw coal feed via a digital twin interface (Hightopo, 2022). For washability specifically, one could imagine an interactive 3D chart where the x-axis is density cut, y-axis is yield, and a third dimension or colour indicates ash – allowing a more intuitive understanding of the trade-offs. While such advanced visualizations are still mostly conceptual, the technology to implement them is largely available through libraries for scientific plotting and VR/AR techniques.

A successful coal washability analysis software should therefore leverage these advances in visualization. It should present complex data in clear, user-friendly formats – for example, dynamic charts, summary tables, and perhaps even suggestions (e.g., “optimal cut density for 10% ash = 1.50, expected yield 70%”). It could also incorporate multi-criteria analysis, letting the user balance parameters (ash vs sulphur vs yield) and see possible outcomes.

## User- Centred Design Considerations

When developing engineering software, especially for a specialized domain like coal beneficiation, it is crucial to follow user-centred design (UCD) principles. User-centred design focuses on the end-users’ needs, making the application intuitive, accessible, and aligned with their workflow (Kigozi, et al., 2024). In the context of a coal washability analysis application, the typical users might be preparation plant engineers, coal quality experts, or researchers. These users often are not software experts, so the interface should be friendly and logical. According to UCD best practices, the design process should involve gathering user requirements, preferences, and pain points, then iteratively prototyping solutions while getting user feedback​ (Kigozi, et al., 2024). For example, users might express the need for easy import of lab data, automatic generation of standard reports, and the flexibility to adjust assumptions. By understanding these needs, the developer can create a tool that genuinely improves the user’s efficiency.

In mining and mineral processing, some software projects have failed in the past because they didn’t align with how engineers work daily. To avoid this, the application should use familiar terminology (specific gravity, yield, ash, etc.), follow normal conventions (for instance, perhaps plotting cumulative floats from low density to high density, as is common in coal reports), and allow outputs to be exported to common formats (Excel, PDF reports, etc.). Visualization components should be customizable because different users might want to see data in different ways. A plant engineer might want a quick summary and a clear marker of the recommended cut density, whereas a researcher might want to see the full curve and underlying raw data points. Interactive elements can enhance usability – for instance, a slider to select cut density and instantly see the resulting product and reject qualities, or the ability to click on the washability curve to get the numerical values. These interactive, feedback-driven designs make the software more engaging and effective as a decision-support tool.

Additionally, the user interface should consider that such software might be used in various settings: from an office desktop with multiple monitors to perhaps a tablet on-site in a lab or plant. This implies a responsive design or at least adaptable layout. The colour scheme should be chosen for clarity (and colour-blind friendliness) when showing multiple curves. If the application involves 3D plant views or similar, it should still be simple to navigate for non-IT personnel.

In summary, the integration of software tools in coal analysis is not just about raw computing power; it’s about creating human-centric tools that enhance understanding and streamline workflows. The trends of digital transformation (like digital twins and AI analytics) provide powerful capabilities, but it is the user-centred design that ensures those capabilities translate into practical benefits. As the industry moves towards more digital, data-driven operations, a well-designed coal washability analysis application can serve as a bridge between complex analytical results and the user’s decision-making process, presenting information in the clearest, most useful manner.

## Impact of Ash Content on Coal Value and Efficiency

Ash content is a critical quality parameter in coal, with significant economic ramifications. Economically, higher ash content generally means lower heating value per unit weight of coal (since ash is essentially non-combustible minerals). Buyers of coal, such as power plants or metallurgical operations, often discount the price of coal with high ash because they are effectively paying for dirt that will end up as waste. A statistical analysis of coal prices in utility contracts found that, all else being equal, coal price correlates positively with energy content and negatively with ash content (Webbink , 1978). In other words, as the percent ash increases, the price a utility is willing to pay per ton decreases, reflecting ash’s dilutive effect on fuel value. Empirical data supports this: one study examining several coals showed that the unit price of coal rose as ash content decreased (and conversely, low-ash premium coals fetched higher prices)​ (Tan, et al., 2010).

From a coal producer’s perspective, removing ash through beneficiation can therefore raise the product’s market value. For example, consider a raw coal with 30% ash that can be washed to 15% ash by discarding 20% of it as refuse; the remaining 80% clean coal might have a significantly higher selling price per ton, often more than compensating for the yield loss. This calculus is at the heart of coal preparation: determining the optimal point where the gain in price (or savings in penalties) outweighs the loss of yield and added processing cost. Beneficiation also can reduce transportation costs – shipping cleaned coal is more efficient than hauling raw coal with a lot of extraneous mineral matter.

In terms of combustion efficiency, ash content plays a detrimental role. High-ash coal lowers the overall thermal efficiency of boilers because a portion of the heat from combustion goes into heating the inert ash to high temperatures and then that heat is lost with the ash in the disposal process. It also means more coal must be fired to achieve the same heat output, which in turn increases fuel handling and grinding requirements. The U.S. Department of Energy notes that moisture and ash in coal negatively impact the plant efficiency and increase *CO₂* emissions per unit of power​ (US Departement of Energy, 2020). Essentially, for each kilowatt-hour of electricity, a high-ash coal will produce more CO₂ because extra coal had to be burned to compensate for the ballast of ash. Additionally, high ash can reduce combustion efficiency by contributing to unburnt carbon in fly ash (if ash particles shield or quench burning coal particles). Modern pulverized coal boilers try to mitigate this with high temperatures and longer residence times, but there is a practical limit – very high ash (say 40% in some low-grade coals) can dramatically reduce boiler efficiency. A specific issue in boilers is that ash can form an insulating layer on heat transfer surfaces (called fouling or slagging when it sinters), which reduces heat absorption by the water/steam and forces a plant either to clean the surfaces or suffer a drop in efficiency​ (US Departement of Energy, 2020). Thus, keeping ash low helps maintain better heat transfer and less frequent cleaning maintenance. In fluidized bed combustors or gasifiers, ash can also affect bed dynamics and throughput.

For metallurgical coal used in coke-making, ash is similarly undesirable. Ash in coke ends up as slag in the blast furnace, requiring flux addition and removal, so steelmakers put tight limits on ash and pay premiums for low ash coking coals. Each 1% reduction in ash can have a sizeable economic benefit in that context due to lower coke rate and less slag handling in the furnace.

## Emissions and Environmental Impacts of Ash

Cleaning coal not only makes economic sense, but it also has environmental benefits. By reducing ash and impurities like sulphur before coal is burned, emissions from combustion are significantly lowered. Washed coal typically produces less particulate matter (PM) emissions because there is simply less ash to become fly ash. As an example, the National Association of Clean Air Agencies has noted that coal washing can dramatically reduce the sulphur and ash content of coal, resulting in a significant reduction in air emissions (NACAA , 2018)​. Lower ash means fewer particulates and less heavy metal content (many toxic elements like arsenic or mercury are concentrated in the mineral matter of coal and report to ash). Power plants with washed coal feed have an easier time meeting emission regulation for PM – the electrostatic precipitators or baghouses have less mass to capture, improving their efficiency and reducing the risk of particulate escape. Less ash also means less fly ash and bottom ash to dispose of after combustion. Coal combustion ash, while often stored or reused (in cement, etc.), contains contaminants (e.g. arsenic, lead, mercury) that can leach if not properly managed (EPA, 2025). By minimizing the ash entering the combustion process (through beneficiation), the total volume of coal ash requiring disposal is reduced.

Moreover, improved combustion efficiency from low-ash coal translates into lower CO₂ emissions per unit of energy. Essentially, if a power station can burn 5% less coal to get the same energy due to higher quality fuel, it will emit ~5% less CO₂ (not to mention NOx and other combustion products). High ash is also correlated with higher levels of certain emissions: for instance, some fraction of nitrogen in coal is in mineral form (ammonium in clays) which contributes to NOx when released; high ash coals might have more such mineral content. Similarly, a portion of coal’s sulphur can be in pyritic form (FeS₂) associated with ash, which oxidizes to SO₂. Washing removes a lot of pyrite (being high density), thus directly cutting potential SO₂ emissions. This pre-combustion removal is far more effective than post-combustion capture in terms of overall environmental impact (though many plants also have scrubbers for sulphur).

That said, coal beneficiation has its own environmental considerations. The process of washing coal produces refuse (reject rock) and tailings (fine waste) that must be managed. If not handled properly, these wastes can cause land and water pollution – for example, refuse rock piles with pyrite can generate acid mine drainage, contaminating waterways with acidic, metal-laden runoff. Fine tailings in slurry form can impound large areas and carry a risk of dam failure if in wet impoundments. There is a strong drive in the industry to minimize these risks by adopting better waste management practices. One approach is dewatering and dry disposal of tailings. Modern filtration technology (like high-pressure filter presses or centrifuges) can dewater fine coal refuse to produce a stackable dry cake (Alam, et al., 2011). The benefits of going to a dry tailing system include significant water recovery (which can be reused in the plant), elimination of seepage into groundwater, and improved stability of the disposed material (Alam, et al., 2011). This reduces the potential for environmental contamination from tailings ponds. Many newer coal prep plants (and mining operations generally) aim for “zero effluent discharge,” meaning all process water is recycled and solid wastes are handled in contained, environmentally stable forms.

Another consideration is the energy and reagents used in beneficiation. DMS uses magnetite, which needs to be mined and processed (though it is recycled many times). Froth flotation, often used for ultra fines, uses chemical reagents that could pose toxicity issues if discharged. However, compared to the environmental benefit of removing pollutants from the coal before burning, these impacts are generally manageable with good practices.

In environmental terms, one can view coal washing as trading a solid waste problem (refuse ash on site*)* for an air pollution problem (fly ash and emissions if unwashed). By concentrating the waste at the mine site (where it can be controlled) rather than dispersing it through the power plant stack, society can better mitigate pollution. Indeed, many countries either mandate coal quality requirements or encourage beneficiation for this reason. For example, coal-fired plants in India and China have faced regulations to use coal below a certain ash percentage for long-distance transport or for feeding power plants, to improve efficiency and reduce emissions at the plant.

From an economic standpoint, environmental factors are increasingly “internalized” – meaning companies have to bear the cost of pollution control and carbon emissions. Using washed coal can reduce these compliance costs. If a power plant can avoid installing an expensive particulate scrubber by using cleaner coal, that is a direct economic incentive. Similarly, lower ash yields less secondary waste (ash handling and disposal is a significant cost for power utilities). A study on the cost of ash in power plants showed that as coal ash content rises, the additional costs (like ash disposal, maintenance, downtime for cleaning, etc.) rise sharply (Tan, et al., 2010). Conversely, investing in coal preparation to keep ash low can be seen as a form of cost avoidance for the end user.

In summary, reducing ash through beneficiation generally yields positive outcomes: higher economic value of the coal, more efficient and cleaner combustion, and lower overall environmental footprint of coal usage. The trade-offs include the cost of washing and the need to manage the concentrated waste. In the context of a coal washability analysis application, these considerations imply that the software could have features to compute not just the yield and quality, but also the economic optimization (e.g., net value yield) and perhaps emissions impact. For instance, the app might allow input of a price penalty per % ash and calculate the optimal cut for maximum revenue or estimate the reduction in CO₂ and SO₂ emissions by washing to a certain ash level. By coupling the technical washability data with economic/environmental models, the tool would be very powerful for decision-making – illustrating the multidisciplinary nature of modern coal engineering where financial, environmental, and technical factors intersect.

## Conclusion

Coal washability analysis is a vital component in the coal beneficiation chain, guiding how we can maximize the recovery of useful coal while rejecting impurities to meet market and environmental requirements. This literature review has covered the foundational principles of Dense Medium Separation in coal processing and highlighted how ongoing innovations, from larger, more efficient cyclones to adaptive control algorithms and even dry separation methods, are pushing the boundaries of what DMS can achieve in terms of efficiency and resource conservation. It was examined how float–sink tests provide the empirical basis for understanding a coal’s cleaning potential, and how the interpretation of washability curves (yield-ash relationships, etc.) sets the theoretical limits for any coal cleaning process. New methodologies such as 3D imaging (RhoVol) and others are emerging to make this analysis faster and safer, reflecting a broader trend of digital transformation in the field.

The integration of software tools into coal analysis is transforming data handling and decision-making. Digital twins, real-time sensors, and advanced data visualization enable engineers to monitor and predict plant performance with unprecedented clarity​ (Zhang, et al., 2021). Emphasizing user-centred design in developing applications ensures that these complex analytics are accessible and actionable to practitioners, bridging the gap between raw data and operational decisions (Kigozi, et al., 2024). Finally, the economic and environmental context provides the rationale for all these efforts: lowering the ash content of coal through beneficiation has clear benefits in terms of fuel value, power plant efficiency, and emissions reduction, aligning with both business interests and regulatory compliance (Tan, et al., 2010) (NACAA , 2018). Of course, responsible management of the resulting waste and the resources used in beneficiation is necessary to truly realize a net positive environmental outcome.

In developing a Coal Washability Analysis Application, one would synthesize these insights. The application can be envisioned as a tool that not only computes and plots washability results, but perhaps also simulates DMS scenarios, integrates with digital plant data, and evaluates the economic/environmental trade-offs of various washing strategies. By grounding the software in the current state of knowledge, as summarized in this review, the resulting application can be both scientifically sound and practically useful. In essence, the trajectory of research and industry practice calls for smarter, more integrated approaches to coal beneficiation, and a well-designed software application is an embodiment of that integration, combining principles of mineral processing, data science, and engineering economics into a single user-friendly platform. Such an application will aid engineers and decision-makers in ensuring that coal, to the extent it continues to be used, is utilized in the cleanest and most efficient way possible, extracting maximum value with minimum environmental impact.

# Chapter 3: Application Overview

This chapter will detail the coal washability web application from an end user’s perspective and explore the key features and functionalities. First, an outline of the technical features of the application will be given from what input data is required to use the application, to what results can be expected. Following this a number of use cases will be explored in order to illustrate the purpose of the application in solving real world problems in move an educational setting and within the operations of a coal washery facility. Final, this chapter will conclude with a look at the user interface of the application and explore the streamline and intuitive design, allowing end users to effortlessly use the application effectively and gain useful information from their coal sink float test data.

## Key Feature

## Purpose and Use Cases

## User Interface

End users can access the coal washability web application via the internet through any web browser of their choice, however, it is recommended to use either Google Chrome or Microsoft Edge, as majority of the testing took place within these web browsers. The web application is made up of a number of routes, accessible via different sub-URLs from the domain. Table X outlines a comprehensive list of the sub-URLs and routes.

|  |  |
| --- | --- |
| **Route** | **Description** |
| / | Landing Page. |
| /calculator | Main page for displaying the washability curves and calculating information related to the coal products required. |
| /about | About page giving user context about the web application and use. |
| /reports | A page to access document related to the Thesis project. |
| /signin | Page for an end user to sign into their personal account. |
| /signup | Page allowing new end users to create an account. |
| /settings | Page for signed in users to make changes to their account. |

### Landing Page

When navigating to the *Home* page of the application via the */*route, the end user will be greeted with the landing page. This page serves as an aesthetic introduction to the coal washability application through a minimalistic design, focused on a user friendly experience, as seen below in Figure X.

A screenshot of a web page

AI-generated content may be incorrect.

Two buttons are placed in the centre of the screen, each allowing the end user to navigate to another page. The first button to the left, the “Get Started” button, will navigate the end user to the /calculator, where they may start using the coal washability calculator immediately. This navigation should be used by an experienced end user who already understands the theory behind coal washability, how coal products can be calculated from washability curves, and how this application can assist in the process, given data provided by the end user.

The second button located directly next to the “Get Started” button, the “Thesis Reports” button, will navigate the end user to the /reports page. If an end user is interested in reading this Thesis report and any other documentation associated with the coal washability application project, before jumping in and utilizing the application, they can access the documentation via this button. This would be ideal for an end user who would like a deeper level understanding the theory behind coal washability, how coal products can be calculated from washability curves, and how this application can assist in the process, given data provided by the end user.

The header of the page, which is common among all the pages, serves as a navigation between the pages. The links in the navigation bar are; *Home* (navigates the end user to the / page), *Calculator* (navigates the end user to the /calculator page), *Reports* (navigates the end user to the /reports page), and *About* (navigates the end user to the /about page). The Home page can also always be accessed by clicking on the icon and “Coal Washability Analysis” title in the left-hand corner of the header. To the right-hand corner of the header, a button with the text “Sign In” will be present to an end user that is not currently signed in to the application. By clicking in this button, the end user will be navigated to the /signin page, where they may either sign into their existing account, or create a new individual account. For a user that is already signed into an existing account on the application, instead of a button with the text “Sign In”, in its place will appear the end users account username, a button in the shape of a gear wheel, and a button with the text “Sign Out”, as seen below in Figure X. The end user may navigate to the /settings page by selecting the gear wheel button, to edit details associated with their individual account. The end user may also sign out of their individual account by selecting the “Sign Out” button.

A screenshot of a computer

AI-generated content may be incorrect.

If the end user requires a brief explanation of the theory behind coal washability, how coal products can be calculated from washability curves, and how this application can assist in the process, given data provided by the end user, they can scroll down on the home page to reveal a brief explanation of the concepts of coal washability, how to calculate coal products from the washability curves, and how the end user can use the application to assist in this process.

A close-up of a graph

AI-generated content may be incorrect.

### Calculator Page

On navigating to the *Calculator* page, the page will appear relatively blank until the end user uploads a dataset. By clicking the “Upload Data” button, the end user will be able to upload a dataset of results collected from a coal sink float test from a given coal source. The initial data points required to be included in the dataset are the density of separation of the data point, the weight percent of the floats for the data point, and the coal ash fraction percent of the floats for the data point.

The “Add New Product” button, which allows end users to calculate different coal products will be disabled until a dataset is uploaded. The Table displaying the calculated coal products will also remain blank until coal products are calculated. The initial state of the Calculator page is shown in Figure X below.

A screenshot of a computer

AI-generated content may be incorrect.

On uploading the coal sink float test dataset to the application, the dataset will be displayed to the end user in a table, allowing user to make changes to the dataset if required. The density of separation, the weight percent of the floats, and the coal ash fraction percent of the floats can all be changed for each datapoint. Each row in the table may be completely removed from the data set by clicking the red “X” button on the right-hand side of the data row. A new data point may also be added to the dataset by clicking the “Add New Row” button the top right-hand corner of the table. This new data row will appear at the bottom of the table and the end user can adjust the density of separation, the weight percent of the floats, and the coal ash fraction percent of the floats for the new data point. If the end user believes they have made a mistake with the data set, they have the ability to remove the dataset from the application and start again, returning them to the initial state of the *Calculator* page. This can be done by clicking on the red “X” button in the top right-hand corner of the table, above the “Add New Row” button.

A screenshot of a computer

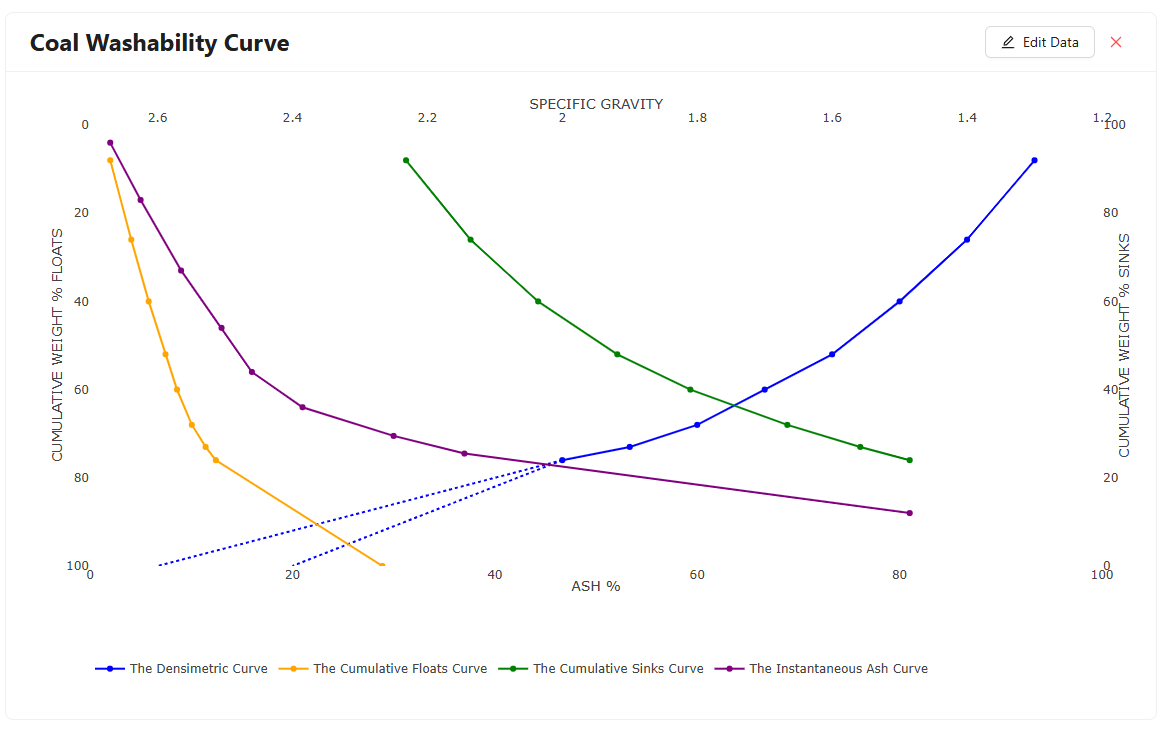
AI-generated content may be incorrect.

When the end user is satisfied with the data set, and are ready to produce the coal washability curves, they can click the “View Plot” button, located in the top right-hand corner of the table. Before producing the coal washability curves, the application with run a number of data validation steps and return an error to the user if the dataset does not pass the validation tests. The first data validation step ensures that each weight percent of the floats data point has a value between 0 and 100. If this is not the case, the error message “Weight % must be between 0 and 100” will be displayed. The second data validation step ensures that each density of separation data point is above 0. If this is not the case, the error message “Density of Separation must be greater than 0” will be displayed. The third data validation step ensures that each coal ash fraction percent of the floats data point has a value between 0 and 100. If this is not the case, the error message “Fraction Ash % must be between 0 and 100” will be displayed. The final data validation step ensures that the weight percent of the floats cumulatively summate to 100. If this is not the case, the error message “Sum of weight % must be 100” will be displayed. The error messages will be displayed above the data table and the end user will be required to make changes to the dataset as they will not be able to proceed to producing the coal washability curves until all errors are resolved.

A screenshot of a computer

AI-generated content may be incorrect.

Once the end user has a dataset that will pass all the data validation steps, they can proceed to producing the coal washability curves by clicking on the “View Plot” button. The coal washability curves will be plotted in the application in place of the data table and the “View Plot” button will be changed to an “Edit Data” button. The coal washability curves plotted consist of; the densiometric curve (blue curve), the cumulative floats curve (yellow curve), the cumulative sinks curve (green curve), and the instantaneous ash curve (purple curve). The curves are plotted on a dual axis graph, with the specific gravity of separation plotted on the top x axis in reverse direction, the ash percentage on the bottom x axis, the cumulative floats weight percentage on the left y axis in reverse direction, and the cumulative sinks weight percentage on the right y axis. The plotted densiometric curve will plot all the values give from the dataset, expect for the last data point. Instead, a range is given for the specific gravity of separation.



Once the initial coal washability curves have been plotted in the application, the “Add New Product” button will turn blue, indicating it is no longer in its disabled mode, and is now clickable. If an end user wants to understand what a coal product of 9% ash would look like with respect to the dataset used (receive information about the products yield and density of separation required), they need only click the “Add New Product” button and update the relevant data. The application will initially give a random ash percentage to the product that can be dynamically updated by the end user. The coal product will receive a name, initially set to “Coal Product 1”, that can be dynamically updated by the end user. The date table, below the coal products, will also now be populated with the data related to “Coal Product 1”, and the final tailings of the coal stream following the washing process. This data table consists of; the name of each coal product, the ash percentage of each product, the yield as a percentage of the total coal input, and the density of separation required to produce the product. Information related to the final tailings will also be included in this data table.

A screenshot of a computer

AI-generated content may be incorrect.

Below the data table, a Sankey plot will also be displayed, reflecting the data included in the data table above. The left side of the Sankey plot will reflect a node of 100% of the raw input coal, and the left side of the Sankey plot will reflect nodes that represent each coal product including the final tailings. The size of each node will reflect the yield percentage of that specific coal product with respect to the raw coal input. The end user can also hover their curser over each node to display all the information as represented in the data table above associated with the specific coal product.



The end user will also have the option to update the name of the coal product they are calculating to something more meaningful than “Coal Product 1”. By clicking on the edit icon next to the coal product name “Coal Product 1”, the text displaying the name will become an editable text input field that can now by dynamically updated by the end user. As the end user adjusts the name of the coal product, all references to the coal product will be dynamically updated also, including the node in the Sankey plot, the dropdown box on the washability curves plot, and the data table located below the ash products. Once the end user is happy with the new name, they can click the edit icon once again and the editable text input field will return to a simple text display of the name. For the example of a 9% ash product, the end user may decide to rename in the relation to the use case of the produce, such as a “Coking Coal”.

A screenshot of a computer

AI-generated content may be incorrect.

Back on the washability curve plots, dashed red lines will now be draw over the plots, illustrating the process taken in calculating both the yield of the coal product, and the density of separation required to produce it. For the 9% ash coal product, the dashed red line will start at the 9% ash point on the x axis and be traced vertically until it reaches the cumulative floats curve. From this point, to find the yield of the product, the dashed red line continues to the left to intercept the cumulative weight percent floats y axis. The value read at this y intercept is the value associated with the yield of the ash product the end user is calculating. For the 9% ash coal product, this value is read to be 62.19% yield. Following this, the dashed red line can be traced horizontally to the right until it intercepts the blue densiometric curve. To get the value of the density of separation required to produce the coal product, the dashed red line is traced vertically up from the blue densiometric curve intercept until the line intercepts the specific gravity of separation x axis at the top of the plot. The value that the dashed red line intercepts at this point will be the density of separation required to produce the coal product. For the 9% ash product, the required density will be 1.73 SG.

A graph of different colored lines

AI-generated content may be incorrect.

With a coal product calculated, the end user may also want information related to the final tailing of the washing process after the product has been produced. To display the final tailing information on the washability curves plot, the end user can use the drop down menu in the top left hand corner to change the selection from their select coal product to the “Final Tailings”. This will redraw the dashed red lines on the washability curves plot, now relating to the process of calculating the yield and the ash percentage of the final tailings instead of the selected coal product. For the final tailings of the 9% ash product example, the dashed red line will start at the value of the specific density of separation that is required for the 9% ash coal product. This value was 1.73 SG on the intercept of the specific gravity of separation x axis at the top of the plot. The dashed red line is then traced vertically downwards from this point until the line intercepts the blue densiometric curve. From this point, to find the yield of the final tailings, the dashed red line continues to the right to intercept the cumulative weight percent sinks y axis. The value read at this y intercept is the value associated with the yield of the final tailings resulting from the coal product the end user is calculating. For the 9% ash coal product, this value is read to be 37.81% yield. Following this, the dashed red line can be traced horizontally to the left until it intercepts the green cumulative sinks curve. At this point, the dashed line can be traced horizontally downwards again until it intercepts the ash percentage x axis. The value at this point will be the average ash percentage of the final tailings. For the final tailings of the 9% ash coal product, the average ash percentage is 61.95%.

A graph of different colored lines

AI-generated content may be incorrect.

With the first coal product calculated and displayed on the washability curves, the end user now has the option to check is an additional product can be made from the tailings, and calculate information related to this second product. To add a new product to the application, the end user ca n again, click the “Add New Product” button. The application will initially give a random ash percentage to the product that can be dynamically updated by the end user. This random ash percentage will be an ash percentage that is actually achievable based on the dataset and first coal product. The coal product will receive a name, initially set to “Coal Product 2”, that can be dynamically updated by the end user in the same way as the first coal product. This new coal product also now be populated within the data table below the coal products with information relating to its yield and density of separation requirements. The final tailing information will now also be updated to reflect the final tails associated with the second coal product, as this second product is made from the first coal products final tailings. For example, the end user may wish to calculate a secondary coal product of 25% ash with the desired application of a steaming coal.

A screenshot of a computer screen

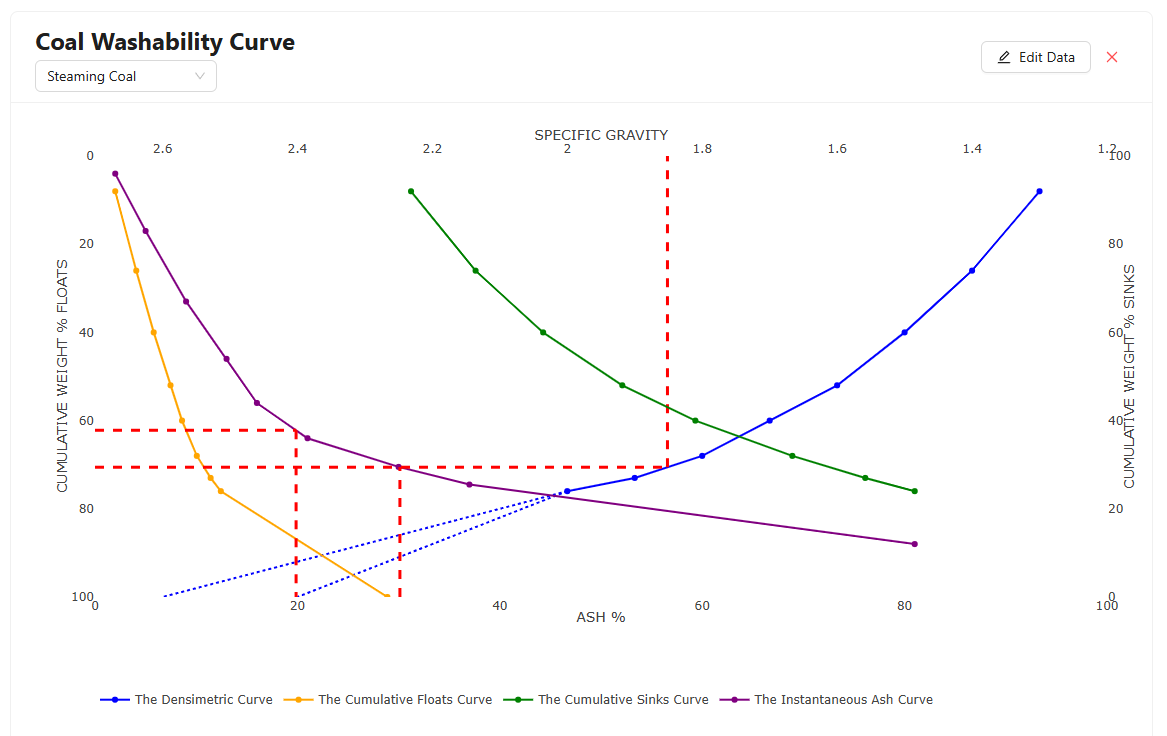
AI-generated content may be incorrect.

Back on the washability curve plots, the dashed red line will now be drawn on the plots to illustrate the process of calculating the relevant information associated with the second coal product, the yield of the coal product, and the density of separation required to produce it. For the 25% “Steaming Coal” product, the dashed red line will start at the first coal products yield, for this example, this was 62.19% yield, the cumulative weight percent floats y axis and be traced horizontally to the right until it reaches the purple instantaneous ash curve. From this point, the dashed red line will be traced vertically downwards to meet the ash percentage x axis. The value at this point will be the lowest ash containing coal product that can be produced from the tailings of the first coal product. For the first coal product (the 9% ash “Coking Coal”) this value is 19.87% ash, thus the lowest ash containing secondary coal product that could be made in this example would be a 19.87% ash coal. As the instantaneous ash curve represents the ash percentage of the coal feed that just floats or just sinks (very close to the separation density), a secondary coal wash yield should approach zero as our desired ash percentage of the secondary coal product approaches zero, as the density of separation will approach the density of separation of the first coal product. Thus, the ash content of a product () produced at a given instantaneous ash point (), will be the average of the minimum ash value () and the instantaneous ash value.

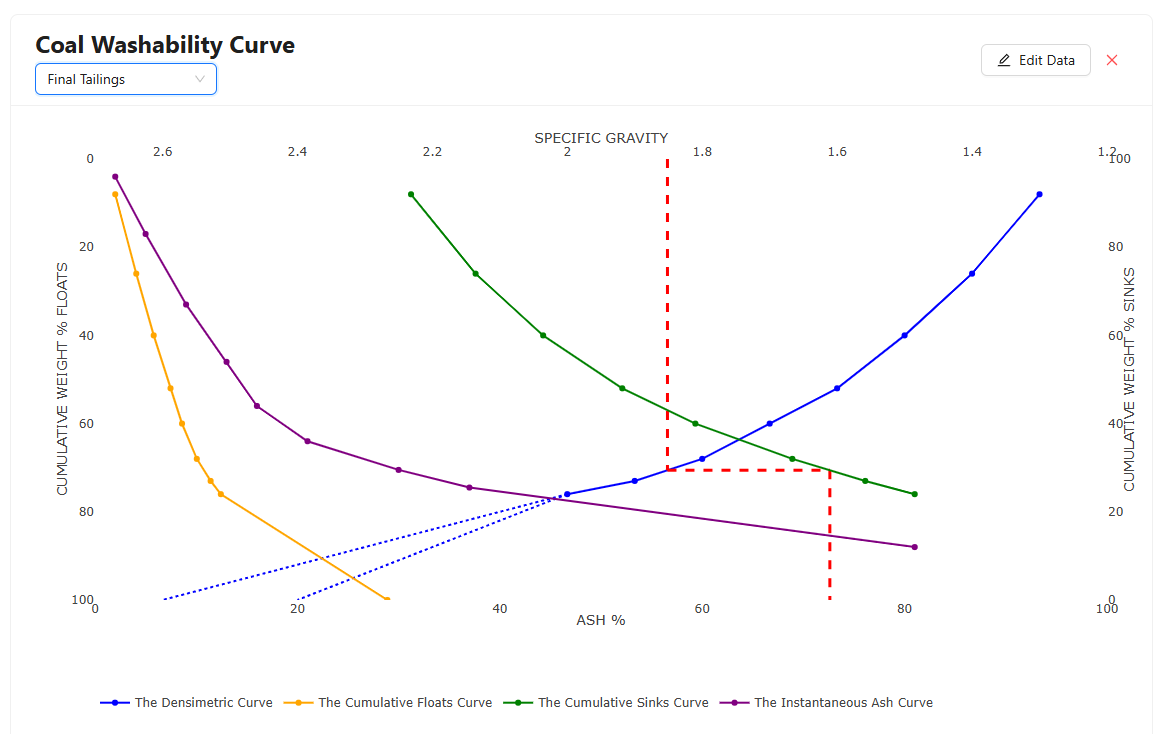
Thus, to find the instantaneous ash value, this equation can be rewritten as:

For the example of a first coal product with an ash of 9% ash and desired second coal product with an ash of 25% ash, the instantaneous ash value is 30.13% ash. This means that the density of separation required to produce the second coal product with 25% ash will be where the coal that either just sinks or just floats (very close to the separation density) have an ash content of 30.13%. To find this density of separation, the dash red line starts at the ash percentage x axis at 30.13% and is traced vertically upwards until it intercepts the purple instantaneous ash curve. From here, the dashed red line can be traced horizontally to the left until it intercepts the cumulative weight percent floats y axis. This value will represent the yield of both coal products up until this point. Thus, to calculate the yield of the second coal product, the yield of the first coal product is simply subtracted from this value. For the example of a first coal product with an ash of 9% ash and desired second coal product with an ash of 25% ash, to total yield of both products is found to be 70.57 yield, and the yield is calculated to be 8.38% yield.

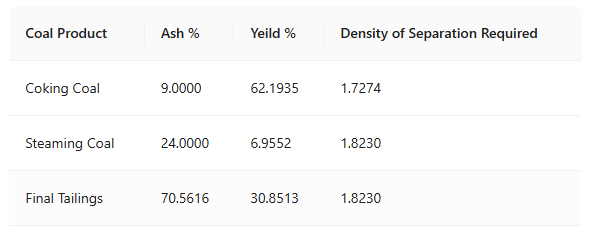
Continuing, the dashed red line is traced horizontally to the right until it intersects the blue densitometric curve. Finally, the dashed red line is trace vertically upwards to meet the specific gravity of separation x axis at the top of the plot. For the example of a first coal product with an ash of 9% ash and desired second coal product with an ash of 25% ash, the density of separation required to product the second coal product is 1.85 SG.



With the secondary coal product information now calculated, the end user can now also view and gather information related to the final tailings from this secondary product. The end user may view the calculation steps for the final tailings yield and ash content by selecting the “Final Tailings” option in the dropdown menu in the top left hand corner of the coal washability curves plot. Similarly to the final tailings of the first coal product, the dashed red line starts at the value of the specific gravity of separation for the previous product at the specific gravity of separation x axis at the top of the washability curves plot. For the example of a first coal product with an ash of 9% ash and second coal product with an ash of 25% ash, the specific gravity of separation for the secondary product is 1.85 SG. From this point the dashed red line is traced vertically downwards until it intercepts the blue densitometric curve. From this point, to find the yield of the final tailings for the secondary coal product, the dashed red line continues to the right to intercept the cumulative weight percent sinks y axis. The value read at this y intercept is the value associated with the yield of the final tailings resulting from the coal product the end user is calculating. For the 25% ash coal product, this value is read to be 29.43% yield. Following this, the dashed red line can be traced horizontally to the left until it intercepts the green cumulative sinks curve. At this point, the dashed line can be traced horizontally downwards again until it intercepts the ash percentage x axis. The value at this point will be the average ash percentage of the final tailings. For the example of a first coal product with an ash of 9% ash and second coal product with an ash of 25% ash, the average ash percentage of the final tailings is 72.61%.



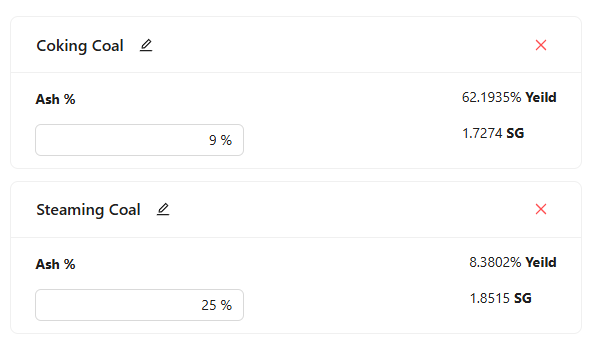
With the information associated with the secondary coal product as well as the information associated with the final tailings of the secondary coal product, the data table, located under the coal products on the right hand side of the screen, will now be populated to reflect and display this information to the end user. Figure X showcases the data table for the calculated information obtained from the example of a first coal product with an ash of 9% ash and second coal product with an ash of 25% ash.



Similarly, the Sankey plot will also now reflect the information associated with the secondary coal product as well as the information associated with the final tailings of the secondary coal product. Figure X showcases the Sankey plot for the calculated information obtained from the example of a first coal product with an ash of 9% ash and second coal product with an ash of 25% ash.



If the end user would like to remove one of the calculated coal products from their calculations and return to a single coal product, they can simply click the red “X” button the top right hand corner of the coal product configuration card.

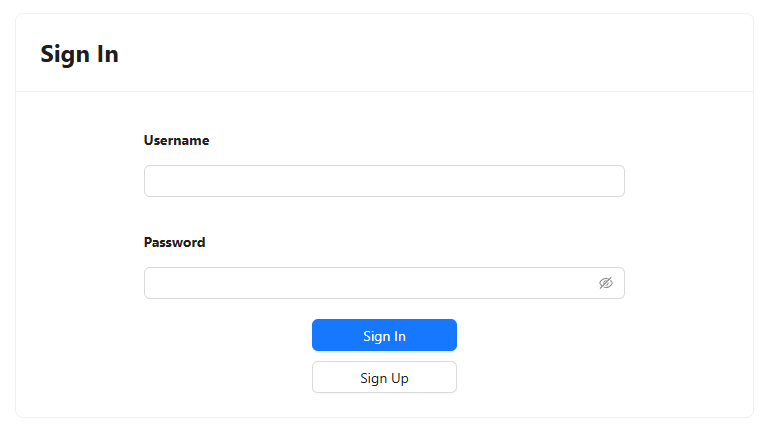


### About Page

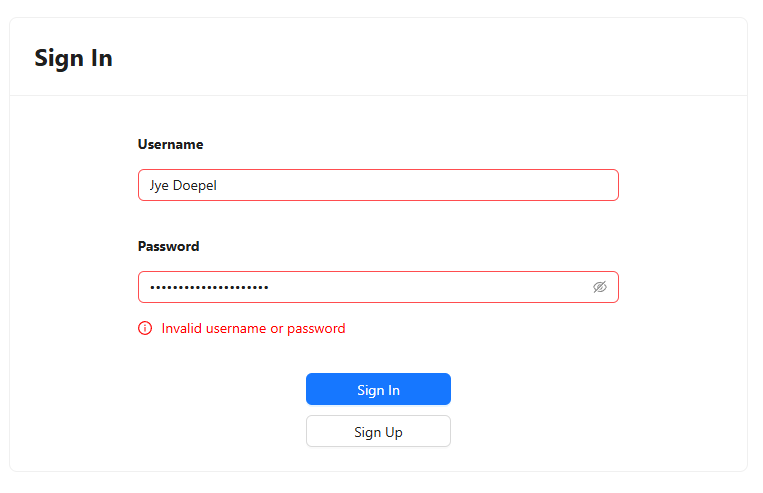
### Reports Page

### Sign In Page

When the end user navigates to the *Sign In* page, they will be presented with a standard user sign in form that they will be required to fill out in order to log in to their unique individual user account for the coal washability application, or they can select the “Sign Up” button to navigate to the *Sign Up* page where the end user will have the ability to create and individual user account for themselves. The end user will be required to enter the credentials of their unique individual user account in order to login to the web application. The credentials required are an account username and corresponding password.



If the end user inputs account credentials that are incorrect and clicks the blue “Sign In” button, the error message “Invalid username or password” will be displayed below the form. Figure X showcases this error message.

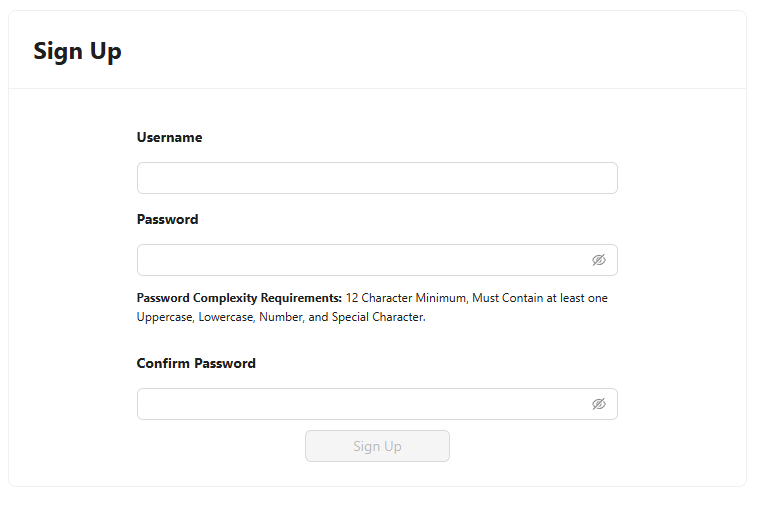


Once the end user enters the correct account credentials and clicks the blue “Sign In” button, the application will navigate to the *calculator* page and the individual user account will be signed in.

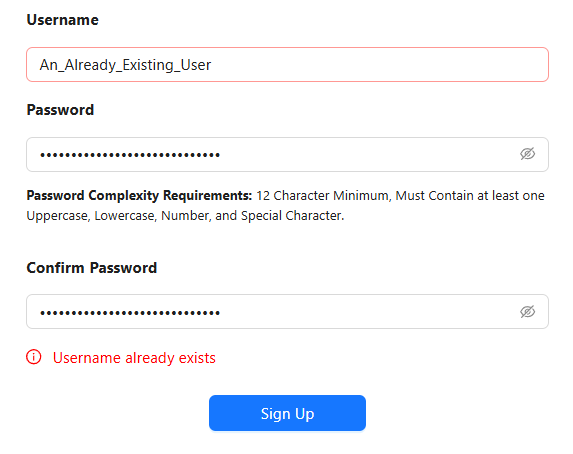
The end user will be required to resign into their account if they leave the application unattended for more than one hour.

### Sign Up Page

On navigating to the *Sign Up* route of the coal washability application, the end user will be presented with a form to create a new user account. If the end user wishes to create a new user account within the coal washability web application, they will be required to create the credentials for this user account within this form. The credentials required are an account username and corresponding password. The end user will be required to enter this password twice, as to confirm that no user error has occurred while the end user enters their password.



When the end user is creating a username for their individual user account for the coal washability web application, the username create must not exist within the web applications database. If the end user enters a username that is associated with an existing user account and clicks the blue “Sign Up” button, an error message reading “Username already exists”.



When creating a password, end users are required to have a password complexity that is in line with ISO 27001. In accordance with the ISO 27001 Information Security Management standard, all end user account passwords must comply with strict complexity requirements to ensure a high level of security across the system (International Organization for Standardization, 2019). These password complexity requirements are designed to protect sensitive information, mitigate the risk of unauthorized access, and uphold the integrity of the web applications digital infrastructure. The policy mandates that every password used for account creation must adhere to the following criteria:

* **Minimum Length:** The password must contain at least 12 characters. This ensures that even the simplest password attempts require more computational power and time to break through brute-force attacks, significantly reducing the risk of compromise.
* **Uppercase Requirement:** The password must include at least one uppercase letter (A–Z). This requirement increases the total number of possible character combinations, thereby enhancing the strength and unpredictability of the password.
* **Lowercase Requirement:** The password must also include at least one lowercase letter (a–z). Combining both uppercase and lowercase characters adds another layer of complexity, making the password harder to guess or decode through dictionary attacks.
* **Numeric Character Requirement:** The password must contain at least one numerical digit (0–9). Including numbers alongside alphabetic characters creates a broader range of possible password permutations and ensures compliance with modern authentication standards.
* **Special Character Requirement:** The password must include at least one special character, such as !, @, #, $, %, ^, &, \*, or similar. Special characters significantly increase password entropy and protect against common hacking techniques, including automated scripts that attempt to guess passwords based on typical user behaviour.

These requirements are not optional and are enforced by the web application at the time of account creation. End users who attempt to register with a password that does not meet all five of the listed complexity conditions will encounter an error message. This error message will be displayed below the two password input boxes and will read “Password does not meet complexity requirements”.

A screenshot of a computer screen

AI-generated content may be incorrect.

It is the end user’s responsibility to ensure that their password meets all complexity requirements before submitting it. This policy has been implemented to reduce security vulnerabilities and to align with best practices outlined in ISO 27001. For further guidance and additional information, end users are encouraged to consult the official Information Security Management Systems Password Complexity Requirements page, where detailed explanations and examples are provided to assist users in creating strong and secure passwords (International Organization for Standardization, 2019). By enforcing this password policy, the coal washability web application reinforces its commitment to robust cybersecurity and regulatory compliance, ensuring that access credentials are resilient against evolving cyber threats. Adhering to these standards not only protects individual end user accounts but also safeguards the entire information ecosystem from potential breaches and data loss incidents.

If all credential requirements are met, the end user can click the blue “Sign Up” button. On the click of this button, an individual user account will be created in the back end of the web application and the end user will be navigated to the *Sign In* page where the they can continue by logging in with these credentials they have just created.

### Settings Page

# Chapter 4: System Architecture and Design

This chapter will outline the technical implementation of the web application with a focus first on the system architecture and development cycle, followed by a detailed discussion of the algorithms and calculations implemented to produce the coal washability curves, calculate information related to the desired coal products, and finally handle the data validation. Finally, this chapter will discuss in detail each element of the technology stack and communication.



## System Architecture and Development Cycle

The system design for the web application utilizes a selection of modern technologies to facilitate the traditional client-server web application architecture. A client-server architecture utilizes an end users’ machine (the client) to ask and receive services from another application running on another machine (the server) (Lile, 1993). The architecture can have huge advantages as it allows computationally expensive and specialised processes to be handled on the server end and return outputs of these processes to the end user to be read and displayed. For this coal washability web application, all calculations and data processing algorithms are handed on the server end, and user input and data display is done on the client’s machine.

Figure x illustrates the key components of the system design and demonstrates the workflow of both an end user interacting with the production environment, and deployment of the development environment application into production.

A diagram of a software company

AI-generated content may be incorrect.

### Development Environment

The development of the coal washability web application was undertaken on a machine running macOS: Sequoia 15.2 running local instances of Python, Node.js, a MySQL server, and a Docker Engine. All code was written in Microsoft’s Visual Studio Code desktop application (Microsoft, 2025). Visual Studio Code is an industry standard tool for web applications development allowing for the utilization of a variety of extensions. Extensions within Visual Studio Code can facilitate connections to services that can streamline both development and deployment of applications. The extension utilized for this project were:

1. **Docker**: This extension allows the user to build a Docker image with the click of a button rather than writing out the full Docker command in a terminal.
2. **GitHub**: The GitHub extension simplifies the source control management of the code base. Commits, Push’s, and Pull’s are streamlined through this extension.
3. **GitHub Copilot**: GitHub Copilot is an AI-powered code completion tool that assists developers by suggesting code snippets, functions, and entire lines based on the context of the current code. It helps speed up the development process by providing context-aware suggestions and reducing the time spent writing boilerplate code. Copilot supports a wide range of programming languages and frameworks, making it a versatile tool for developers working on various aspects of a project.

When developing a new application, it is always useful to setup a development environment that is unique to the application. Development environment typically utilizes a unique instance of the packages and libraries used by the application and are typically governed by configuration files. This is useful, as an identical development environment can be set up on another machine given the same configuration file, eliminating the phrase “Well it works on my computer”. Conda was utilized for the Python development environment management for the development of this project (Anaconda, 2025). It allows developers to create isolated environments with specific versions of libraries, tools, and Python itself. This helped ensure that the project was consistent across different machines, reducing conflicts between dependencies and making it easier to manage multiple other projects with different requirements on the same machine. For the React.js server, Node.js’ default package manager, npm (Node Package Manager), was used to manage the development environment (npm, 2025). Npm works in a similar way to Conda’s python environment manager. The key difference between npm and Conda is npm is a project dependency manager, whereas Conda is an environment manager, theoretically used across multiple projects. The use of Conda and npm allowed for a solid backbone for the development of the project through their development environment management abilities.

A local instance of a MySQL server was installed on the development machine to be used as a development database for the development of the coal washability web application (Oracle, 2025). MySQL is a relational database management system (RDBMS), meaning it stores date in relational models using tables. RDBMSs are widely used in various applications, especially for managing structured data with well-defined relationships. The MySQL server listening port was configured to the industry standard and default MySQL server port, 3306 (MySQL, 2025). Communication between the Python server and the MySQL database was easily managed with the MySQL Python connector package where SQL statements could be easily executed with minimal lines of code, streamlining the development process (MySQL, 2025).

The application server of the project should be viewed as two isolated and distinct components:

1. **The Front End**: The front-end server component refers to the server that feeds the end user with a dynamically updated HTML page via HTTP requests. This server is implemented using the JavaScript framework React.js.
2. **The Back End**: The back-end server component refers to the server that feed the front-end server with data via HTTP requests and API. This server is implemented using the Python framework FastAPI.

### Docker Deployment

To move the application from the development environment to the production environment, each server component was built into an image using the Docker engine. Docker is a platform that streamlines the deployment of applications and management though containerization. Containers are very similar to virtual machines in the sense that they are isolated environments running their on operating systems within a machine level operating system, however, they are typically much more lightweight than a full virtual machine, only having the necessary components to run the application within the container. Docker images are a template for a container to be created from. A dockerfile is used to instruct the Docker engine what the container dependencies will be.

For the backend server dockerfile, python:3.11-slim was first installed. This is a lightweight instance of python that does not come with Python’s full package list and libraries installed. This allows for optimization of the image size as the full package list was not required to run the application. Following this installation, the project backend server source code is copied to the image and the environment configuration requirements file is run to install all the required packages that are not natively installed with python:3.11-slim. Finally, the image is asked to expose the port 8000 to the Docker engine, as this the port the backend server listens for HTTP requests from the frontend server.

For the frontend server dockerfile, node:18-alpine is first installed. This is similar to the python:3.11-slim install on the backend Docker image, as node:18-alpine is lightweight instance of node without the full package list installed by default. The alpine variant is a version of the Node.js image that is based on Alpine Linux, a lightweight and security-oriented Linux distribution. It is significantly smaller than the standard Node.js image because it has fewer dependencies, making it ideal for use in production environments where minimizing image size is a priority. Following this installation, the project frontend server source code is copied to the image and the environment configuration package file is run to install all the required packages that are not natively installed with node:18-alpine. Additionally, nginx:alpine is installed to act as middleware in the product environment between the end user and the React.js application. React applications, when built for production, generate static assets like HTML, CSS, and JavaScript files. nginx:alpine is an efficient web server for serving these static files. Finally, the image is asked to expose the port 80 to the Docker engine, as this the port the frontend server listens for HTTP requests from end users.

When the server images have been built onto the development machines Docker engine, they are then ready to be pushed to a container registry that can be accessed by the production machine. A container registry is a repository for docker images that can be used as an intermediate step to move images from one machine to another (e.g. development machine to production machine). An Azure container registry service was used for this project.

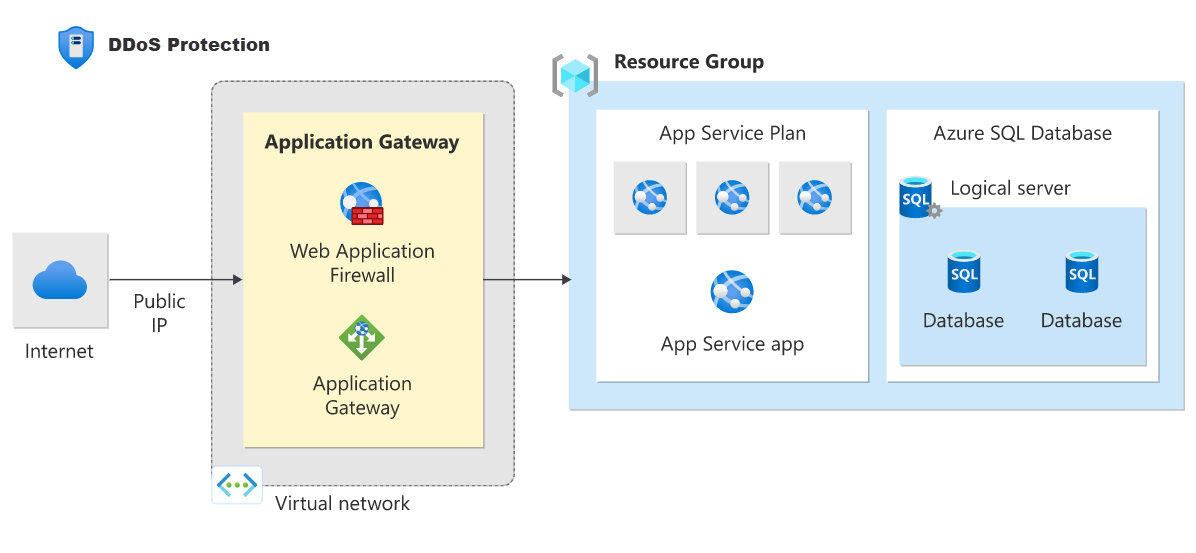
### Production System

The production system of the coal washability web application runs on a non-graphical Debian Linux operating system (Debian, 2025), on an Azure Cloud Standard B1s Virtual Machine in Australia East (Microsoft, 2025). Debian is a popular, open-source Linux distribution known for its stability and reliability. It is one of the oldest and most widely used distributions in the Linux ecosystem, and it serves as the foundation for many other distributions that can be used for everything from desktops to servers and embedded systems. Running a non-graphical Debian distribution can also reduce the computational load on the virtual machine, reducing the machine size required to facilitate the application. Azures Standard B1s Virtual Machine size is part of Azures Bv1 size’s general purpose series. Bv1 Virtual machine sizes are designed to provide cost efficient machines for applications that do not require continuous high vCPU performance, however, they have the ability provide bursts of high performance when required, which is ideal for a web application hosting, optimizing vCPU usage when end users are making requests of the server (Microsoft, 2025). The standard B1s Virtual Machine used for this project has one Intel Xeon Platinum vCPU with hyper threaded logic cores (Intel, 2025). These hyper threaded logic cores are key to handling multiple end users at the same time, as multiple threads can be utilized to handle multiple requests at once, whilst limiting the impact on the vCPU performance.

Azures cloud computing Virtual Machine services offer a range of benefits compared to traditional *on prem* server machines including high availability and reliability with a guaranteed uptimes of 99.99% (Microsoft, 2025), and the flexibility and customization of operating systems and Virtual Machine sizes, however, an additional key benefit utilized for this project was Azures built in security infrastructure (Microsoft, 2025). Azures Network Security Groups were key to controlling inbound and outbound traffic to the applications running on the virtual machine (Microsoft, 2025). Azures Network Security Groups act as a virtual firewall, managing traffic based on rules that define which ports, IP addresses, and protocols are allowed or denied, that can all be controlled through Azures online portal rather than within the Debian Linux operating system itself. Table X outlines the details of the allowed rules in the Network Security Group used for this project.

|  |  |  |  |
| --- | --- | --- | --- |
| **Port** | **Protocol** | **Source** | **Description** |
| 22 | TCP | Administrator IP | Allows system Administrator to use SSH to access the virtual machine remotely. |
| 3000 | HTTP | Virtual Machine IP | FastAPI backend container listening port. |
| 8000 | HTTP | Any | Nginx frontend container listening port. |
| 5000 | HTTP | Administrator IP | Docker Container Registry listening port. |
| 3306 | TCP | Virtual Machine IP | MySQL server listening port. |

An additional key part of Azures built in security infrastructure is DDoS (Distributed Denial of Service) attack protection (Microsoft, 2025). DDoS attacks are orchestrated attacks whereby multiple systems flood a server with traffic, resulting in the server being unable to handle requests from real end users attempting to use the application (Douligeris & Mitrokotsa, 2024). These attacks can have major impacts on the uptime and reliability of an applications servers. Azures DDoS protection is designed to protect infrastructure within the Azure ecosystem with Automatic traffic monitoring, detecting suspicious activity and denying requests before they reach the virtual machine.



The Debian Linux operating system was installed with a MySQL server to be used as the production database for the coal washability application, and a Docker Engine in order to run the application containers. The MySQL server was setup to mirror the development databases schema for production to reduce any conflict between and production and development environments. With the application images stored on the Azure container registry, the images could be pulled to Docker engine running on the virtual machine. Once on the machine, two containers were deployed in the Docker engine, one running the back-end server, and the other running the front-end server. The back-end server was configured to map port 8000 on the container to port 8000 on the virtual machine in order for the front-end server to access the end points as governed by the rules in the Network Security Group used for this project. Similarly, the front-end server was configured to map port 80 on the container to port 3000 on the virtual machine to facilitate requestion from end users. With the application containers running on the Docker Engine and the MySQL server running and mirroring the schema of the development database, the Coal washability web application was now accessible via the internet to end users from the public IP address of the Virtual Machine.

## Algorithms and Calculations

Through leveraging the client-server design for the system architecture, all of the algorithms and calculations were performed on the back-end server side of the application. This approach provides a number of benefits for both the performance and stability of the application across different end users and the computational capacities of their machines. The following section will outline in detail the algorithms and calculations implemented in this project to:

1. Handle Initial Input Coal Sink/Float Data.
2. Validate Input Coal Sink/Float Data.
3. Calculate information related to required coal products.
4. Calculate information related to final tailing after producing coal products.
5. Authentication of end users on the application.

### Handling Initial Input Coal Sink/Float Data.

End users are required to upload their coal sink/float dataset through the web application in either a CSV file format or a JSON file format. These files require, at minimum, three columns of data:

1. *Density of Separation* or *A*
2. *Weight %* or *B*
3. *Fraction Ash %* or *E*

The data file is sent from the front-end server to the FastAPI back-end server via a HTTP request to the */data\_upload* end point in the form of a binary request body. The binary request body is decoded to a *utf-8* string and is read into a panda DataFrame. Pandas is a powerful Python library used for data manipulation and analysis, providing flexible data structures for handling and analysing structured data (Pandas, 2025). A validation step is then initiated to ensure that the correct minimum columns are present within the DataFrame. If the correct columns are not present, an error will be returned to the front-end server. Following this date validation step, the data will be sorted in ascending order by the *Density of Separation* column. From here all other relevant data points can be calculated in order to produce the washability curves.

First the weight percent of each fraction [C] should be calculated. This will be the weight of the fraction the floats divided by the total weight of the fraction (Tolhurst, 2024).

data['C'] = data['B'] / data['B'].sum() \* 100

The cumulative weight percentage [D] can then be calculated through a sum of the weight percent fractions (Tolhurst, 2024).

data['D'] = data['C'].cumsum()

To calculate the weight of the ash fraction as a percentage of the total weight [F], the fraction weight is multiplied by the ash percentage in the fraction (Tolhurst, 2024).

data['F'] = data['C'] \* data['E'] / 100

The cumulative weight percentage of the ash [G] can then be calculated through a sum of the weight percent fractions (Tolhurst, 2024).

data['G'] = data['F'].cumsum()

The cumulative floats ash [H] can then be calculated by dividing the cumulative weight percentage of the ash by the weight of the ash fraction as a percentage of the total weight (Tolhurst, 2024).

data['H'] = data['G'] / data['D'] \* 100

The sink weight percentage of ash [I] is calculated from the total ash percentage minus the ash in the cumulative floats (Tolhurst, 2024).

data['I'] = data['F'].sum() - data['G']

The cumulative sinks weight percent [J] can be calculated by subtracting the cumulative weight percentage by 100 (Tolhurst, 2024).

data['J'] = 100 - data['D']

The cumulative sinks ash percentage [K] can be calculated by dividing the sink weight percentage of ash by the cumulative sinks weight percent (Tolhurst, 2024).

data['K'] = data['I'] / data['J'] \* 100

The heaviest floats percentage with that ash percentage [M] can be calculated by (Tolhurst, 2024):

data['M'] = None

for i, row in data.iterrows():

    if i == 0:

        data['M'][i] = data['C'][i] / 2

    else:

        data['M'][i] = data['D'][i-1] + data['C'][i] / 2

The dataset is the converted from a Pandas DataFrame to a JSON string. This JSON string is then returned to via HTTP response to the front-end server.

### Validating Input Coal Sink/Float Data

Before producing to calculating the full dataset required to product the coal washability curves, the backend server runs a number of data validation steps and return an array containing the error/s to the front end server if the dataset does not pass an element of the validation tests. The first data validation step ensures that each weight percentage of the floats data point has a value between 0 and 100. Each element of the column “B” (*Weight % of floats*) from the data set is checked by looping over the data set. If the individual element does not pass the test, the error “Weight % must be between 0 and 100” is added to the errors array that will be returned to the front end server.

errors = []

for i, row in data.iterrows():

    if row['B'] < 0 and row['B'] > 100:

        errors.append("Weight % must be between 0 and 100.")

The second data validation step ensures that each density of separation data point is above 0. Each element of the column “A” (*Density of Separation*) from the data set is checked by looping over the data set. If the individual element does not pass the test, the error “Density of Separation must be greater than 0” is added to the errors array that will be returned to the front end server.

for i, row in data.iterrows():

    if row['A'] < 0:

        errors.append("Density of Separation must be greater than 0.")

The third data validation step ensures that each coal ash fraction percent of the floats data point has a value between 0 and 100. Each element of the column “E” (*Fraction Ash % of floats*) from the data set is checked by looping over the data set. If the individual element does not pass the test, the error “Fraction Ash % must be between 0 and 100” is added to the errors array that will be returned to the front end server.

for i, row in data.iterrows():

    if row['E'] < 0 and row['E'] > 100:

        errors.append("Fraction Ash % must be between 0 and 100.")

The final data validation step ensures that the weight percent of the floats cumulatively summate to 100. Utilizing the a pandas DataFrame to store the dataset allows for an easy cumulative sum with a single line of code. If the dataset does not pass this test, the error “Fraction Ash % must be between 0 and 100” is added to the errors array that will be returned to the front end server.

floats\_cum\_sum = data['B'].cumsum()

if floats\_cum\_sum != 100:

    errors.append("Sum of weight % must be 100.")

Following the completion of all the data validation checks, if the errors array contains any elements, the array is striped to leave only the unique elements. This needs to be done as the same error may appear multiple time in the errors array if the validation step fails on multiple elements of the same test. For example, if values less then 0 appeared in column “B” twice, the element “Weight % must be between 0 and 100” would appear in the errors array twice. The striped errors array is then returned.

if len(errors) > 0:

    errors = list(set(errors))

return errors

### Calculating Information Related to Required Coal Products

On the data validation step and the full coal washability dataset has been calculated, the allowable ash range of the first coal product can be calculated. The range for the first coal product minimum value is the minimum value of the “H” (The cumulative floats ash) column in the dataset and the maximum value is the second largest value of the “H” (The cumulative floats ash) column in the dataset. The allowable product ranges are stored in an array with the index of the array representing the coal product.

product\_ranges = [

    {

        'max': sorted(set(data['H']), reverse=True)[-1],

        'min': min(data['H'])

    }

]

The full coal washability dataset and the product ranges are then returned to the front end server application where the end user can begin to calculate the first coal product within the given product ash range. Once the end user has requested the calculation of a coal product with a given ash percentage the request is sent back through the same function as about to calculate the first coal product range (this is done every time, as the dataset is editable by the end user at any time and can lead to these values changing). The request has the following schema:

{

    "data": JSON object of washability data,

    "products": array of Ash percentage for each desired product,

}

If the “*products”* array is not empty, the application will loop over each element of the “*products”* array, calculation; the yield of the product as a percentage of the total raw coal, the density of separation required to product the coal product, and the coordinates of the dashed red calculation lines to be displayed over the coal washability curves on the end users system. For the first coal product in the “*products”* array, the desired ash percentage and the coal washability dataset are passed into the *first\_product()* function, returning; the coordinates of the dashed red calculation lines (*traces*), the density of the separation required for the products (*sg*), the yield of the product (*clean\_yeild*), and the allowable ash range of any further coal products (*next\_product\_range*).

for index, product in enumerate(coal\_products):

    if index == 0:

        traces, sg, clean\_yeild, next\_product\_range = first\_product(product, data)

The coal washability dataset is inherently discrete. That is, values for ash and yield are measured at specific SG intervals and do not span a continuous domain. This presents a challenge when end users specify a desired product with an ash percentage that do not directly exist in the dataset. To address this, the function checks whether the end user specified ash value (ash) exists within column H. If not, the function inserts a new row and interpolates the corresponding values of specific gravity (A) and cumulative yield (D) using linear interpolation. The interpolation is calculated as:

Where and are the two known ash values that bond the desired ash value and , and , are the specific gravity of separation and the yield of floats at those bounds. This interpolation assumes linear behaviour between data points, a reasonable assumption given the typically smooth trend of washability curves in narrow density intervals.

if data[data['H'] == ash].shape[0] > 0:

    new\_row\_index = data[data['H'] == ash].index[0]

else:

    new\_rows = pd.DataFrame({'H': [ash]})

    data = pd.concat([data, new\_rows], ignore\_index=True)

    data = data.sort\_values(by='H')

    data = data.reset\_index(drop=True)

    new\_row\_index = data[data['A'].isna()].index[0]

    if new\_row\_index > 0:

        percent\_increase = (data['H'][new\_row\_index] - data['H'][new\_row\_index-1]) / (data['H'][new\_row\_index+1] - data['H'][new\_row\_index-1])

        for col in ['A', 'D']:

            data[col][new\_row\_index] = (data[col][new\_row\_index+1] - data[col][new\_row\_index-1]) \* percent\_increase + data[col][new\_row\_index-1]

Once the target ash row is inserted and interpolated, the function constructs three Plotly-compatible line traces to be rendered on the graph:

* **Vertical Ash Line:** Drawn at the X-coordinate of the user-defined ash target, from 100% to the interpolated yield value. This marks the ash boundary selected by the user.
* **Vertical SG Line:** Plotted at the interpolated SG value, descending from the yield % to the X-axis (0%). This visually shows the required density of separation.
* **Horizontal Yield Line:** Extends from the origin to the intersection with the SG axis or the ash axis, visually representing the recovered product’s yield at the chosen cut-point.

Each line is styled with a red dashed pattern to differentiate it from the primary washability curves. These lines provide a clear graphical reference for the trade-off between ash quality and recovery, a key decision point in coal preparation plant design and control.

traces = [

    {

        'x': [ash, ash],

        'y': [100, np.float64(data['D'][new\_row\_index])]

    },

    {

        'x': [np.float64(data['A'][new\_row\_index]), np.float64(data['A'][new\_row\_index])],

        'y': [np.float64(data['D'][new\_row\_index]), 0]

    },

    {

        'x': [0, np.float64(data['H'][new\_row\_index]) if np.float64(data['H'][new\_row\_index]) > (np.float64(2.7)-np.float64(data['A'][new\_row\_index]))/sg\_range\*100 else (np.float64(2.7)-np.float64(data['A'][new\_row\_index]))/sg\_range\*100],

        'y': [np.float64(data['D'][new\_row\_index]), np.float64(data['D'][new\_row\_index])]

    }

]

Following the primary product calculation, the function attempts to estimate a feasible ash range for a second coal product. This is accomplished by reverse mapping the yield to its associated ash content using the “M” and “E” columns. If the desired yield does not exist in column “M”, the function again interpolates the value of ash (E) at that yield, providing the lower bound of the next product’s ash range. To approximate the upper bound, the function inserts another interpolated row at the second-highest cumulative yield value, calculates the ash content at that yield, and computes the midpoint between the two ash values:

This midpoint serves two purposes. It provides a reasonable estimate for the next product’s ash target, which can be pre-populated in the interface to guide the user. It ensures that ash ranges between products remain non-overlapping and logically sequential, reducing the chance of ambiguous product definition or visual clutter on the washability curve. This technique enables the application to facilitate rapid definition of multi-product coal recovery strategies, while still honouring the constraints and resolution of the original float-sink data.

if data[data['M'] == clean\_yeild].shape[0] > 0:

        new\_row\_index = data[data['M'] == clean\_yeild].index[0]

        remove\_row = False

else:

    new\_rows = pd.DataFrame({'M': [clean\_yeild]})

    data = pd.concat([data, new\_rows], ignore\_index=True)

    data = data.sort\_values(by='M')

    data = data.reset\_index(drop=True)

    new\_row\_index = data[data['A'].isna()].index[0]

if new\_row\_index > 0:

    percent\_increase = (data['M'][new\_row\_index] - data['M'][new\_row\_index-1]) / (data['M'][new\_row\_index+1] - data['M'][new\_row\_index-1])

    for col in ['E']:

        data[col][new\_row\_index] = (data[col][new\_row\_index+1] - data[col][new\_row\_index-1]) \* percent\_increase + data[col][new\_row\_index-1]

min\_ash\_of\_next\_product = np.float64(data['E'][new\_row\_index])

new\_rows = pd.DataFrame({'M': [data['D'].nlargest(2).iloc[-1]]})

data = pd.concat([data, new\_rows], ignore\_index=True)

data = data.sort\_values(by='M')

data = data.reset\_index(drop=True)

new\_row\_index = data[data['A'].isna()].index[0]

percent\_increase = (data['M'][new\_row\_index] - data['M'][new\_row\_index-1]) / (data['M'][new\_row\_index+1] - data['M'][new\_row\_index-1])

for col in ['E']:

    data[col][new\_row\_index] = (data[col][new\_row\_index+1] - data[col][new\_row\_index-1]) \* percent\_increase + data[col][new\_row\_index-1]

max\_ash = 100-(data['A'].nlargest(2).iloc[-1] - (data['A'].min() - 0.1))/sg\_range\*100

max\_ash\_of\_next\_product = (min\_ash\_of\_next\_product + data['E'][new\_row\_index])/2

next\_product\_range = {'max': max\_ash\_of\_next\_product, 'min': min\_ash\_of\_next\_product}

Finally, the calculated coordinates of the dashed red calculation lines (*traces*), the density of the separation required for the products (*sg*), the yield of the product (*clean\_yeild*), and the allowable ash range of any further coal products (*next\_product\_range*) will be returned from the function.

For the second coal product in the “*products”* array, the desired ash percentage, the coal washability dataset, and the yield of the previous coal product are passed into the *second\_product()* function, returning; the coordinates of the dashed red calculation lines (*traces*), the density of the separation required for the products (*sg*), the yield of the product (*clean\_yeild*), and the allowable ash range of any further coal products (*next\_product\_range*).

for index, product in enumerate(coal\_products):

    if index == 1:

        traces, sg, clean\_yeild, next\_product\_range = second\_product(product, data, previous\_yeild)

The function first retrieves the ash content corresponding to the previously recovered yield. Since the coal washability dataset is discrete, the yield value may not match exactly. Thus, the function interpolates within the M (yield) vs. E (ash) domain:

Where becomes the starting ash content of the second product and is the previous products yield. This interpolation defines where the second product begins in ash space.

Next, the function calculates the required ash cut point to symmetrically centre the second product about the end user’s desired ash target. It uses:

This expression ensures that the second product is centred at the desired ash value, with one end fixed at the end of the first product. It is based on the midpoint identity:

The function then interpolates again, this time in the E and M domain to find the cumulative yield corresponding to target ash. This determines the upper cumulative yield after the second cut.

After determining the cumulative yield required for the second product, the function interpolates a third time, now using D (cumulative floats yield) and A (density of separation) to determine the specific gravity at which this second cut occurs:

This cut point becomes the density of separation required to product the second coal product with the target ash content. The yield calculated thus far includes recovery from both the first and second products. The function subtracts the first yield to isolate the incremental recovery of the second product:

The function then generates five red dashed line traces to illustrate the bounds of the second coal product:

* **First yield line (horizontal):** from 0% to the ash boundary of the first product at the yield level.
* **First vertical line:** up from the yield of the first product to 100%.
* **Second vertical line:** down from 100% to the new cumulative yield at the second product’s ash cut point.
* **Second yield line (horizontal):** from origin to ash or SG position at the new yield.
* **SG line (vertical):** marks the density of separation for the second cut.

These visual cues allow users to see where the second product fits within the washability domain and how much material can be economically recovered without breaching ash quality constraints.

Finally, the calculated coordinates of the dashed red calculation lines (traces), the density of the separation required for the products (sg), the yield of the product (clean\_yeild), and the allowable ash range of any further coal products (next\_product\_range) will be returned from the function.

With all the desired information for each coal product calculated, the data is organised into a JSON response for the front end server to revive. This response consists of:

* *Data*: This is the original coal washability dataset.
* *Products*: This is the original array of desired ash contents for each coal product, where the index corresponding to the specific coal product.
* *Sg:* An array of the specific gravity of separations required to product each coal product, where the index corresponding to the specific coal product.
* *Clean\_yeild:* An array of the yield for each coal product, where the index corresponding to the specific coal product.
* *Table\_data:* A JSON object that summarises the calculated data from each coal product in a format that can be easily consumed by the React.js Ant Design component *<Table/>.*
* *Options:* An array of the end user defined names for each coal product, where the index corresponding to the specific coal product.
* *Product\_ranges:* An array of each coal products allowable ash content range, where the index corresponding to the specific coal product.

response = {

'data',

'products',

'sg',

'clean\_yeild',

'table\_data',

'options',

'product\_ranges',

}

### Calculating Information Related to Final Tailing After Producing Coal Products

The *final\_tailings()* function models and visualizes this residual material within the washability curve interface. It calculates the ash content and yield of the tailings based on the last product's cut-point and generates red dashed line traces to visually represent the tailings on the washability plot. The function accepts two parameters:

* *data*: a dictionary-like structure containing the float-sink test data. It is immediately converted into a pandas DataFrame for tabular manipulation.
* *sg*: the specific gravity (SG) at which the last coal product was separated.

The coal washability dataset is inherently discrete, often sampled at specific SG intervals. If the provided density of separation value does not match an existing density in the dataset, the function must interpolate the corresponding values required to calculate tailings. To do this a new row with the target density of separation (A = sg) is appended to the dataset. Next, the dataset is sorted by specific gravity of separation. Finally, linear interpolation is performed on columns “D” (cumulative floats yield), “J” (cumulative sinks yield), and “K” (cumulative sinks ash).

The function then generates three red dashed line traces to illustrate the bounds of the final tailing’s region:

* **SG line (vertical):** from the X-axis (0%) up to the cumulative yield at the final separation specific gravity, plotted on the SG axis.
* **Tailings yield line (horizontal):** from the SG's ash-axis position across to the interpolated ash content of the tailings at a constant yield level.
* **Tailings ash line (vertical):** up from the cumulative yield level to 100%, marking the boundary of unrecovered material.

These visual markers delineate the region of the washability curve that represents discarded material, allowing end users to assess how much feed remains outside the recovery window defined by the clean coal products. This not only completes the mass balance but also supports decision-making around product specification versus yield trade-offs.

### Password Cryptography and Authentication of End Users on the Application

User authentication in the coal washability web application is implemented using industry standard security protocols and cryptographic methods, including bcrypt hashing for password storage and JWT (JSON Web Token) generation for stateless session management. This design conforms with modern cybersecurity best practices and aligns with ISO/IEC 27001 standards on user credential protection and password complexity enforcement (International Organization for Standardization, 2019).

Storing passwords in plain text poses a significant security risk. Instead, the application uses the bcrypt algorithm via the passlib library to securely hash user passwords before they are stored in the database. The bcrypt algorithm is a robust and adaptive password hashing algorithm that leverages the Blowfish cipher, salted input, and exponential cost scaling to provide secure, future-proof password storage (Provos & Mazieres, 1999). Its resistance to brute-force, dictionary, and precomputation attacks makes it highly suitable for securing user credentials in modern web applications (Provos & Mazieres, 1999). The *hash\_password()* function invokes *pwd\_context.hash(password)* to generate a salted hash using bcrypt. This method introduces a cryptographic salt, a random value added to each password prior to hashing, to defend against rainbow table attacks and ensure that identical passwords result in different hashes (Sriramya & Karthika, 2015). Bcrypt is also computationally expensive by design, offering resistance to brute-force attacks by increasing hash computation time (Sriramya & Karthika, 2015). When a user attempts to sign in, their entered plain-text password is compared against the stored hash. This ensures that the plain password is hashed with the same algorithm and the comparison is done in a secure, timing-attack-resistant manner.

The function *is\_password\_strong()* enforces password complexity rules aligned with the ISO/IEC 27001 security standard (International Organization for Standardization, 2019). The policy mandates that every password used for account creation must adhere to the following criteria:

* **Minimum Length:** The password must contain at least 12 characters. This ensures that even the simplest password attempts require more computational power and time to break through brute-force attacks, significantly reducing the risk of compromise.
* **Uppercase Requirement:** The password must include at least one uppercase letter (A–Z). This requirement increases the total number of possible character combinations, thereby enhancing the strength and unpredictability of the password.
* **Lowercase Requirement:** The password must also include at least one lowercase letter (a–z). Combining both uppercase and lowercase characters adds another layer of complexity, making the password harder to guess or decode through dictionary attacks.
* **Numeric Character Requirement:** The password must contain at least one numerical digit (0–9). Including numbers alongside alphabetic characters creates a broader range of possible password permutations and ensures compliance with modern authentication standards.
* **Special Character Requirement:** The password must include at least one special character, such as !, @, #, $, %, ^, &, \*, or similar. Special characters significantly increase password entropy and protect against common hacking techniques, including automated scripts that attempt to guess passwords based on typical user behaviour.

This ensures that passwords are resilient to dictionary and brute-force attacks and meets organizational compliance and auditing requirements (International Organization for Standardization, 2019). If the password does not meet these criteria during the signup process, the server returns a 400 Bad Request error with the error message “Password does not meet complexity requirements”.

The *signup()* function handles user registration. Its workflow is as follows:

1. The password is first validated for strength.
2. It is then hashed using bcrypt.
3. The hashed password and username are inserted into the user table of the MySQL database.
4. If the username already exists (e.g., violates a unique constraint), a controlled exception is raised to prevent information leakage about existing accounts.

Importantly, at no point is the plaintext password stored, either in the database or in application logs, thus maintaining confidentiality.

After successful login, the application implements stateless session management using JWTs (JSON Web Tokens). JWTs are cryptographically signed tokens that encode user identity and expiration metadata, and are used instead of server-side sessions (Adam, et al., 2020). The *create\_jwt\_token(username)* function generates the token with the following structure:

payload = {

'sub': username,

'exp': datetime.datetime.now() + datetime.timedelta(hours=1)

}

Token = jwt.encode(payload, SECRET\_KEY, algorithm=”HS256”)

Where *sub* (subject) contains the username, representing the authenticated identity and *exp* (expiration) adds a token validity window (1 hour in this implementation). The token is signed using HMAC-SHA256 with a server-side *SECRET\_KEY* stored in environment variables via *.env* to ensure cryptographic integrity. When returned to the client, this token is stored on the frontend and used in subsequent API requests. This enables the application to remain stateless, and there is no need to store session information on the server side. At each request, the server can decode the JWT, verify its signature and expiration, and extract the user’s identity from the token’s payload.

The *signin()* function validates user login as follows:

1. The end user provided username is used to query the database for the corresponding password hash.
2. The provided plain-text password is verified against the stored hash using bcrypt.
3. If validation succeeds, a JWT is generated and returned to the client.
4. If validation fails, a controlled HTTP exception is raised with a generic error message to prevent user enumeration.

The application ensures that authentication failures do not disclose whether the username or password was incorrect, enhancing security by resisting username harvesting attempts.

## Technology Stack

### Frontend

The front end of the coal washability web application is built using Meta’s JavaScript framework, React. React is a popular open-source JavaScript library used for building user interfaces for applications where data changes frequently (Gackenheimer, 2015). Developed by Meta, React simplifies the process of creating interactive UIs by allowing developers to build reusable components that manage their own state (Gackenheimer, 2015). React uses a virtual DOM to update and render components, improving performance by minimizing direct manipulation of the real DOM (Gackenheimer, 2015). React’s component-based architecture and wide range of third party libraries make it powerful and flexible for creating modern web applications.

On top of React, two third party applications were utilized for in the development of the coal washability web application. These being Ant Design, and Plotly. Ant Design (AntD) is a comprehensive React UI library developed by Alibaba, designed to create enterprise-level web applications with consistency and efficiency (Ant Group, 2025). It offers a wide range of high quality, reusable components such as buttons, tables, forms, modals, and navigation elements, all styled according to a clean, modern design system (Ant Group, 2025). Ant Design promotes a design philosophy centred on natural interaction, minimalism, and clear hierarchy, which helps maintain visual and functional consistency across applications (Ant Group, 2025). Its components are highly customizable and come with built-in internationalization support, responsive design principles, and strong JavaScript integration (Ant Group, 2025). With excellent documentation, Ant Design streamlines the development process and enables rapid prototyping and deployment of complex interfaces, making it especially popular data-driven applications.

Additionally, Plotly was utilised within the front end of the application to display the coal washability curve to the end user. Plotly is a React wrapper for the Plotly.js charting library, enabling the creation of interactive and responsive data visualizations within React applications (Li & Bilal, 2021). This integration combines Plotly’s powerful charting capabilities, including support for bar charts, scatter plots, heatmaps, and 3D visualizations, with React’s declarative UI model (Li & Bilal, 2021). Plotly supports real-time updates, user interactions (like zooming, hovering, and clicking), and responsiveness to browser resizing (Li & Bilal, 2021). It is especially well-suited for analytics tools and scientific applications where high interactivity and customization are required. Charts can also be styled dynamically and updated using React state or hooks.

Front\_end/

├── index.html

├── package.json

├── vite.config.js

├── src/

│ ├── index.css

│ ├── main.jsx

│ ├── App.jsx

│ ├── components/

│ │ ├── AppHeader.jsx

│ ├── pages/

│ │ ├── AppAbout.jsx

│ │ ├── AppCalculator.jsx

│ │ ├── AppHome.jsx

│ │ ├── AppReports.jsx

│ │ ├── AppSettings.jsx

│ │ ├── AppSignIn.jsx

│ │ ├── AppSignUp.jsx

React usings a utilises a single html page that is dynamically updated through JavaScript running on the end users web browser (Gackenheimer, 2015). The *index.html* file contains the single html page used by the coal washability React app. This file has a very simple structure, it links to the CSS style sheet *index.css,* containing the all the CSS styling for the application, and a root div html element. This root div html element is what React uses to as its virtual DOM in order to dynamically update the web page of the application. The *main.jsx* file is the starting point of the React application. Again, this file has a very simple structure, identifying the root div html element in the *index.html* file, and setting this as the virtual DOM for dynamical updating using the *createRoot* React function. The *App* component from the *App.jsx* file is then rendered inside the root div html element.

import App from './App.jsx'

createRoot(document.getElementById('root')).render(

    <App />

)

The *App* component from the *App.jsx* file servers as a container for the application and contains the React Router. A React Router is used to facilitate different pages within the coal washability web application what can be accessible via URL navigation instead of continuously updating a single React component (Duldulao & Cabagnot, 2021). Each page is governed by its own individual React component and the React Router simply renders a given pages React component when the end user navigates to the page via the URL. The *Router* component is used a wrapper for the application, will all other components falling within this component. This tells the React that the application is a React Router application. Within this component the routable pages are wrapped in the *Routes* component. This component defines all the different routes of the web application. Each route is defined using the *Route* component, with the component of the specific page passed into the specific *Route* component as the *element*. The specific sub-URL path of the page is passed into the *Route* component as the *path.* The one component of the page that remains rendered in the virtual DOM, regardless of the navigation route, it the *AppHeader* Reactcomponent.

With in the App component, the JWT token is also checked by getting the value of the “*access\_token”* within the end users’ browsers local storage and stored with the “*token*” React state. In React, state is a built-in object that stores dynamic data in a component and determines how that component behaves and renders. When the state changes, React automatically re-renders the component to reflect the new data, enabling interactive and responsive user interfaces. Each React component can have its own state, managed using the “*useState”* hook in functional components. State can hold any type of data, such as strings, numbers, arrays, or objects, and is used to track inputs, toggles, fetched data, and more.

import AppHeader from './components/AppHeader'

import AppHome from "./pages/AppHome";

import AppCalculator from "./pages/AppCalculator";

import AppAbout from "./pages/AppAbout";

import AppReports from "./pages/AppReports";

import AppSignIn from "./pages/AppSignIn";

import AppSignUp from "./pages/AppSignUp";

import AppSettings from "./pages/AppSettings";

function App() {

 const [token, set\_token] = useState(localStorage.getItem('access\_token'));

 return (

   <Router>

       <AppHeader ip={ip} token={token} set\_token={set\_token}/>

       <main style={{marginTop: '5%'}}>

         <Routes>

           <Route path="/" element={<AppHome />} />

           <Route path="/calculator" element={<AppCalculator ip={ip}/>} />

           <Route path="/about" element={<AppAbout />} />

           <Route path="/reports" element={<AppReports ip={ip}/>} />

           <Route path="/signin" element={<AppSignIn ip={ip} set\_token={set\_token}/>} />

           <Route path="/signup" element={<AppSignUp ip={ip}/>} />

           <Route path="/settings" element={<AppSettings ip={ip}/>} />

         </Routes>

       </main>

   </Router>

 )

}

The *AppHeader* component from the *AppHeader.jsx* file serves as a header and the main navigation for the coal washability web application. The component is rendered with the *App* component outside of the *Routes* component and is therefore always rendered within the virtual DOM regardless of URL navigation. The navigation bar within the header utilizes an Ant Design *Menu* component to produce the horizontal navigation menu. A list of the pages; *Home, Calculator, Reports,* and *About*, are based into the *Menu* component as the variable *items*. Each element of the array consists of a *label* (text to be displayed in the menu), and a *url* (sub-URL navigation link). The *onClick* function within the *Menu* component will navigate the end user to the sub-URL associated with that specific element of the *items* array.

<Menu onClick={(e) => { navigate(e.url) }} selectedKeys={[appRoute]} mode="horizontal" items={items} />

Each *App* rerender (e.g. page change) the *AppHeader* will authenticate the current JWT token saved in the *token* state of the *App* component (if one exists, this is only true if the end user is signed into an account) by calling the /authenticate/*token* backend end point. The authentication process is dependent on a valid JWT token that has been correctly signed by the backend server and not past the expiration timestamp. If this is correct, the /authenticate/*token* backend end point will return a response containing the end users account username and the expiration timestamp. The end users account username will then be updated in the *user* React state. If the authentication is not successful, the application will remove the token within the end users web browsers “*access\_token”* key, as well as clear the *user* React state.

useEffect(() => {

    let isMounted = true;

    async function fetchData() {

        const response = await fetch(`${ip}/authenticate/${token}`);

        const data = await response.json();

        console.log(data);

        if (response.status == 200) {

            if (isMounted) {

                set\_user(data.sub);

            }

        }

        else {

            if (isMounted) {

                set\_token(null);

                set\_user(null);

            }

        }

    }

    if (token !== null) {

        fetchData();

    }

    return () => { isMounted = false; };

}, [token]);

On the right hand side of the *AppHeader* component contains a button that will navigate the end user to the *SignIn* route. If the authentication was successful, and the *user* React state stores the username of the end user account, this button will be replaced with; the username of the end users account, a new button with will sign the end user out of their account by clearing the *user* React state and removing the token within the end users web browsers “*access\_token”* key, and a button that will navigate the end user to the *Setting* route.

{user != null ?

    <div style={{ display: 'flex', alignItems: 'center', justifyContent: 'space-between' }}>

        <p style={{ margin: '0px' }}>

            {user}

        </p>

        <SettingOutlined onClick={() => navigate('/settings')} className="no\_border\_button" style={{cursor: 'pointer'}}/>

    </div>

:

    <></>

}

<p className="no\_border\_button" style={{ marginLeft: 'auto', cursor: 'pointer', marginTop: '5px', textAlign: 'right' }} onClick={goToSignIn}>

    <b>{user == null ? "Sign In" : "Sign Out"}</b>

</p>

function goToSignIn() {

    if (user != null) {

        localStorage.removeItem('access\_token');

        set\_token(null);

        set\_user(null);

    } else {

        setCappRoute('/signin');

        navigate('/signin');

    }

}

The *AppCalculator* component from the AppCalculator.jsx file serves as the core interface for the coal washability analysis in the web application. This component governs a number of React states, including:

* *Upload\_data:* Stores the raw sink float dataset uploaded by the end user.
* *coalProducts:* an array of defined coal products with the schema:

coalProduct = {

‘edit\_name’,

'name',

'id',

'ash',

}

* *plotData:* the full calculated coal washability dataset, formatted for Plotly integration.
* *­selected\_product:* tracks the active selected coal product for displaying the calculation lines on the plot.
* *Editing:* toggles between plot view and raw date editing view.
* *plotElement:* coal washability Plotly component.
* *Sankey\_plot:* Sankey Plotly component.
* *Table\_data:* stores the dataset related to the calculated information related to each desired coal product.

The web applications handles the upload of the end user’s dataset through the utilization of an Ant Design *Upload* component. This *Upload* component streamlines the API call to the backend end point */data\_upload* and handles the io stream of the uploaded dataset file*.* The file and API responses is then handled by the *upload\_data* function.

<Upload {...upload\_data }>

    <Button icon={<UploadOutlined />}>Upload Data</Button>

</Upload>

 const upload\_data = {

   name: 'file',

   action: `${ip}/data\_upload`,

   headers: {

     authorization: 'authorization-text',

   },

   onChange(info) {

     if (info.file.status !== 'uploading') {

       console.log(info.file, info.fileList);

     }

     if (info.file.status === 'done') {

       message.success(`${info.file.name} file uploaded successfully`);

       setCoalProducts([]);

       setselected\_product(null);

       settable\_data([]);

       setplotData(info.file.response);

       set\_uploaded\_data(info.file.response.original\_data);

       console.log(info.file.response.original\_data);

       set\_data\_error\_messages([]);

       setEditing(true);

     } else if (info.file.status === 'error') {

       message.error(`${info.file.name} file upload failed.`);

     }

   },

 };

On a successful response from the */data\_upload* backend server end point, a JSON response object is returned to React application with the following key data points.

* *Data:* this is an Object of the full, calculated coal washability dataset.
* *Clean\_yeild:* An array of the containing the yield as a percentage of the raw coal feed of each specific coal product. Before a coal product has been calculated this array will be empty.
* *Option:* An array of the names of all the desired coal products including the final tailings. Before a coal product has been calculated this array will be empty.
* *Product\_ranges:* an array of Objects with the keys ‘*min’* and ‘*max’,* corresponding to the minimum and maximum possible ash contents of each coal product and next possible coal product. Before any coal product has been calculated this array will only contain the minimum and maximum possible ash contents for the first coal product.
* *Products*: An array of the coordinates for constructing the dashed red calculation lines for each product on the coal product, including the final tailings. Before a coal product has been calculated this array will be empty.
* *Sg*: An array of the specific gravities of separation needed to produce each desired coal product. Before a coal product has been calculated this array will be empty.
* *Table\_data*: A summary of all the coal products, including the final tailings, containing the yield, specific gravity of separation required to produce, and ash content. This is in a format that is easily interpretable with an Ant Design *Table* component.

Each React state is then updated to reflect the data received from this API response. At this point, most of the react states will remain as empty arrays, as no coal products have been attempted to be calculated, however, if the end user has been working on an old dataset, and is now uploading a new dataset, all of the information in the React states will now be over written with the information related to the most recently uploaded dataset and the old data will be lost.

### Backend

### Database

# Chapter 5: Conclusion

This thesis has presented the development of a Coal Washability Analysis Web Application, a digital tool designed to streamline, automate, and enhance the interpretation of coal washability data derived from sink float testing. Grounded in the principles of dense medium separation (DMS), the application addresses key challenges in coal preparation, including the complexity of manual calculations, risk of human error, and barriers to learning for new users. By digitizing the process and providing dynamic visualizations of washability curves, the tool empowers both students and industry professionals to make informed, data-driven decisions about coal beneficiation.

The application was built using a modern and scalable technology stack, combining React.js for an intuitive front-end experience and FastAPI for efficient backend processing. Core scientific computations were handled using Python libraries like NumPy and Pandas, while Plotly enabled interactive data visualization. The result is a responsive, user centred platform capable of calculating product yields, determining separation densities, and displaying results through washability curves and Sankey diagrams. Its flexibility and usability make it suitable for educational institutions, research environments, and coal preparation facilities alike.

Beyond technical implementation, this project reflects a broader shift toward digital transformation in the coal industry. By integrating computational tools into traditional workflows, coal beneficiation can become more efficient, accurate, and environmentally conscious. This thesis contributes to that vision, offering a tool that not only simplifies analysis but also helps reduce emissions and improve the economic value of coal products through optimized decision-making.

In conclusion, the Coal Washability Analysis Web Application exemplifies how engineering principles, digital technologies, and user focused design can intersect to solve practical challenges in mineral processing. With further refinement and integration into industry workflows, such tools can play a pivotal role in shaping the future of clean, efficient coal utilization.

# References

Alam, N., Ozdemir, O., Hampton, M. A. & Nguyen, A. V., 2011. Dewatering of coal plant tailings: Flocculation followed by filtration. *Fuel,* 90(1), pp. 26-35.

Anaconda, 2025. *Conda.* [Online]   
Available at: https://anaconda.org/anaconda/conda

Bhagwat, S. B., 2009. *Estimation of Coal-Cleaning Costs: A Spreadsheet-Based Interactive Software for Use in the Estimation of Economically Recoverable Coal,* Illinois: U.S. Geological Survey Professional Paper.

Botlhoko, S., Campbell, Q. P., Roux, M. L. & Nakhaei, F., 2022. Washability Analysis of Coal Using RhoVol: A Novel 3D Image-based Method. *Mineral Processing and Extractive Metallurgy Review,* 44(2), pp. 1-13.

Burton, M. et al., 1991. The economic impact of modern dense medium systems. *Minerals Engineering,* 4(3-4), pp. 225-243.

Cheng, W. et al., 2018. Washability and Distribution Behaviors of Trace Elements of a High-Sulfur Coal, SW Guizhou, China. *Minerals,* Volume 59.

Cuiping, W. et al., 2023. Study on plotting of coal washability curve using the software Python. *Coal Preparation Technology,* Volume 51, pp. 79-83.

Dai, W., 2020. Model-data-based switching adaptive control for dense medium separation in coal beneficiation. *Control Engineering Practice,* 98(8).

Daley, P., Mendoza, K. L. M., Sabogal, L. M. & Williams, O. S. A., 2022. Upgrading of Low-Grade Colombian Coals via Low-Cost and Sustainable Calcium Nitrate Dense Media Separation. *American Chemical Society.*

David, 2017. *Shaker Table Washability Curves.* [Online]   
Available at: https://www.911metallurgist.com/blog/shaker-table-washability-curves/#:~:text=made%20at%201,the%20separation%20is%20made%20is  
[Accessed 2025].

David, 2019. *Dense Medium Cyclone.* [Online]   
Available at: https://www.911metallurgist.com/blog/dense-medium-cyclone/#:~:text=quite%20recently%2C%20the%20largest%20dense,Such%20a  
[Accessed 2025].

Debian, 2025. *Home Page.* [Online]   
Available at: https://www.debian.org/

Douligeris, C. & Mitrokotsa, A., 2024. DDoS attacks and defense mechanisms: classification and state-of-the-art. *Computer Networks,* 4 April, 44(5), pp. 643-666.

EPA, 2025. *Coal Ash Basics.* [Online]   
Available at: https://www.epa.gov/coalash/coal-ash-basics#:~:text=Coal%20Ash%20Basics%20,waterways%2C%20ground%20water%2C%20drinking  
[Accessed 2025].

Gackenheimer, C., 2015. What Is React?. In: *Introduction to React.* s.l.:Apress, pp. 1-20.

Gailer, B. et al., 2024. *GoalGuru: A React- and FastAPI-based Cloud Application for Predicting Soccer Games Outcome,* Amberg, Germany: CyberLytics-Lab.

Galvin, K., 2025. *Options for Washability Analysis of Coal - A Literature Review.* [Online]   
Available at: https://www.acarp.com.au/abstracts.aspx?repId=C14006#:~:text=The%20fundamental%20result%20obtained%20from,at%20a%20new%20washability%20method

Groppo, J., 2017. An introduction to the nature of coal. *Coal Combustion Products,* pp. 3-20.

Hightopo, 2022. *Coal Washing | Coal Preparation Plant | Web 3D Digital Twin | 3D Visualization.* [Online]   
Available at: https://www.youtube.com/watch?v=CXumMsza-Ik#:~:text=,3D%20Visualization  
[Accessed 2025].

Hughes, N., Roux, M. l., Campbell, Q. P. & Nakhaei, F., 2024. A review of the dry methods available for coal beneficiation. *Minerals Engineering,* Volume 216.

Intel, 2025. *Get Faster Performance For Many Demanding Business Applications.* [Online]   
Available at: https://www.intel.com/content/www/us/en/architecture-and-technology/hyper-threading/hyper-threading-technology.html

International Organization for Standardization, 2019. *ISO/IEC 27701:2019 – Security Techniques – Extension to ISO/IEC 27001 and ISO/IEC 27002 for Privacy Information Management – Requirements and Guidelines,* Geneva: s.n.

Kentucky Geological Survey, 2024. *Float-Sink (Washability) Test.* [Online]   
Available at: https://www.uky.edu/KGS/coal/coal-analyses-float-sink.php#:~:text=Because%20coal%20is%20less%20dense,fraction%20at%20that%20liquid's%20density.  
[Accessed 2024].

Kentucky, U. o., 2025. *Float-Sink (Washability) Test.* [Online]   
Available at: https://www.uky.edu/KGS/coal/coal-analyses-float-sink.php#:~:text=of%20densities,weighed%2C%20and%20its%20density%20and

Kigozi, J. et al., 2024. Human-centered design in the development of innovative technology for the food processing medium, small and micro enterprises (MSMEs) in Uganda. *Journal of Engineering Research.*

Lile, E. A., 1993. Client/server architecture: A brief overview. *Journal of Systems Management.*

Lin, C. L., Luttrel, G. H., Adel, G. T. & Miller, J. D., 1998. *DEVELOPMENT OF AN ON-LINE COAL WASHABILITY ANALYZER,* Morgantown, West Virginia: Federal Energy Technology Center.

Lin, S., 2023. *What Drives Coal's Price?.* [Online]   
Available at: https://legacy.maqro.com.au/blog/coal/what-drives-coals-price/  
[Accessed 2024].

Li, R. & Bilal, U., 2021. Interactive web-based data visualization with R, plotly, and shiny. *Biometrics,* 77(2), pp. 776-777.

Maptek, 2020. *Coal Washability in Vulcan 2020.* [Online]   
Available at: https://www.maptek.com/video/coal-washability-in-vulcan-2020/#:~:text=Coal%20Washability%20in%20Vulcan%202020  
[Accessed 2025].

Meyer, E. & Craig, I., 2014. Coal dense medium separation dynamic and steady-state modelling for process control. *Minerals Engineering,* Volume 65, pp. 98-108.

Micromine, 2018. *Dealing with the complexity of Coal data with Geobank 2018.* [Online]   
Available at: https://www.micromine.com/dealing-with-the-complexity-of-coal-data-with-geobank-2018/#:~:text=2018%20www,storage%20of%20beneficiation%20analytical%20data  
[Accessed 2025].

Microsoft, 2025. *Azure infrastructure security.* [Online]   
Available at: https://learn.microsoft.com/en-us/azure/security/fundamentals/infrastructure

Microsoft, 2025. *Bv1 sizes series.* [Online]   
Available at: https://learn.microsoft.com/en-us/azure/virtual-machines/sizes/general-purpose/bv1-series?tabs=sizebasic

Microsoft, 2025. *Home Page.* [Online]   
Available at: https://code.visualstudio.com/

Microsoft, 2025. *Network security groups.* [Online]   
Available at: https://learn.microsoft.com/en-us/azure/virtual-network/network-security-groups-overview

Microsoft, 2025. *Service Level Agreement for Microsoft Online Services April 1, 2025,* s.l.: Microsoft.

Microsoft, 2025. *What is Azure DDoS Protection?.* [Online]   
Available at: https://learn.microsoft.com/en-us/azure/ddos-protection/ddos-protection-overview

Minerals Council of Australia, 2024. *Minerals Council of Australia About.* [Online]   
Available at: https://minerals.org.au/about/mining-facts/mineral-coal/

MySQL, 2025. *MySQL Connector/Python Developer Guide.* [Online]   
Available at: https://dev.mysql.com/doc/connector-python/en/

MySQL, 2025. *MySQL Port Reference Tables.* [Online]   
Available at: https://dev.mysql.com/doc/mysql-port-reference/en/mysql-port-reference-tables.html

NACAA , 2018. *Improve Coal Quality,* s.l.: NACAA .

npm, 2025. *Home Page.* [Online]   
Available at: https://www.npmjs.com/

Oracle, 2025. *Home Page.* [Online]   
Available at: https://www.mysql.com/

Pandas, 2025. *About pandas.* [Online]   
Available at: https://pandas.pydata.org/about/

Phengsaart, T., 2022. *onventional and Recent Advances in Gravity Separation Technologies for Coal Cleaning: A Systematic and Critical Review.* [Online]   
Available at: https://ssrn.com/abstract=4249751

Ranjani, J., Sheela, A. & Meena, K. P., 2019. *Combination of NumPy, SciPy and Matplotlib/Pylab -a good alternative methodology to MATLAB - A Comparative analysis.* Chennai, India, International Conference on Innovations in Information and Communication Technology.

SAHU, B., 2013. *STUDY OF WASHABILITY CHARACTERISTICS OF,* s.l.: DEPARTMENT OF MINING ENGINEERING.

Tan, H. et al., 2010. Decision Making on Most Economical Coal for Coal-Fired Power Plants Under Fluctuating Coal Prices. *International Journal of Coal Preparation and Utilization,* 31(5), pp. 273-288.

Tolhurst, R., 2024. *Minerals Benefication - Lecture 6 Dense Medium Separation, (DMS) Sink/Float Notes.* s.l.:University of Wollongong.

US Departement of Energy, 2020. *Global Thermal Coal Attributes,* s.l.: US Departement of Energy.

Webbink , D. W., 1978. *THE EFFECT OF SULFUR BTU AND ASH CONTENT ON THE PRICE OF ELECTRIC UTILITY COAL,* Washington: BUREAU OF ECONOMICS FEDERAL TRADE COMMISSION.

Wei, L., 1998. Mechanism and application of coal dry beneficiation with air-dense medium fluidized bed. *Journal of Central South University of Technology,* Volume 5, pp. 100-103.

Ze-lin, Z. et al., 2011. A study on Fast Predicting the Washability Curve of Coal. *Procedia Environmental Sciences,* Volume 11, pp. 1580-1584.

Zhang, Q., Chen, H., Zhu, Y. & Wang, J., 2021. Digital Twin Based on 3D Visualization and Computer Image Recognition Technology in Coal Preparation Plant. *Journal of Physics: Conference Series.*

# Appendices