International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol.10, No.2, 2023, pp.96-105.

ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

Empowering Users with An Intuitive Web App for Air Compressor Troubleshooting and Optimization

Dr.N. Baggyalakshmi [a*], B.C. Danuja [b], Dr.R. Revathi [c]

[a]Assistant Professor, Department of Computer Science, PSGR Krishnammal College for Women, India.*
E-mail: baggyanethra@gmail.com
[b] Student, BSC. Computer Science, PSGR Krishnammal College for Women, India.
E-mail:dhanuja1709@gmail.com
[c] Assistant Professor, Department of Computer Science, Karpagam Academy of Higher Education, India.
E-mail:ravathilakshay@gmail.com

Abstract

An E-Support system is an online system that provides users with support and assistance for various products or services. These systems are available 24/7, which means that customers can get the help they need at any time. The project is to implement and help people through online by developing an online website for compressor. This can be accessed online from any part of the world. It is a user - friendly website and was developed in visual studio code using java script, html and css. The app can provide step-by-step guidance to diagnose and resolve air compressor issues quickly. Users can avoid spending hours searching for solutions online or flipping through manuals, which saves valuable time and reduces downtime for their equipment. Quick problem identification and resolution mean less downtime for the air compressor. In industrial or commercial settings, minimizing equipment downtime is crucial for maintaining productivity and meeting deadlines. The customers can use this user-friendly website for checking compressors. The main objective of this web app is to provide users with a user-friendly, efficient, and effective tool for diagnosing and resolving issues related to air compressors. The app aims to help users quickly identify the root cause of problems and offer appropriate solutions to restore the compressor's functionality.

Keywords: Air Compressor, Web App, E-Support, Optimization.

DOI:10.71086/IAJSE/V10I2/IAJSE1013
Received: 17 March 2023/Revised: 21 April 2023/Accepted: 12 May 2023/Published: 29 June 2023

Introduction

This web app can be used for troubleshooting problems with compressors efficiently and provides a set of guided suggestions to engineers on field. The main page contains fault diagnostics, unit converters and air system calculators. It can help users identify and fix common air compressor problems on their own. This can save money on service calls or professional maintenance, especially for minor issues that don't require specialized skills [1].

Web apps are accessible from various devices, making it easy for users to access the troubleshooting tool from their desktop computers, laptops, tablets, or smartphones. They can access the app whenever they encounter issues with their air compressors, even while on the go. Manufacturers or maintenance teams can update the app in real-time to reflect new troubleshooting techniques or address emerging issues. Users will always have access to the most up-to-date information [2].

The app can help prevent inexperienced users from attempting incorrect troubleshooting steps that might cause further damage to the air compressor. By providing reliable guidance, the app minimizes the risk of errors during the repair process. The web app can serve as a centralized hub for troubleshooting information related to specific air compressor models and brands. This ensures consistency in troubleshooting methods and promotes best practices [3].

For businesses with multiple locations or remote sites, the web app can act as a virtual support tool, enabling off-site technicians or supervisors to assist local staff in troubleshooting and repairs. The app can be used as a training tool for new maintenance personnel or operators. It can help them understand the various components of the air compressor and develop troubleshooting skills [4].

Making sure downstream equipment that uses compressed air continues to work properly is crucial for the client. As a result, saving time and money requires savvy troubleshooting. We can guess what the issue is by watching the symptoms closely. Proper data collection, verification, and ascertainment are prerequisites to drawing any firm conclusions [5].

Table 1: Cause and Suggested Solution

CAUSE	SUGGESTED SOLUTION
Cooler fins clogged externally	Clean the cooler externally as recommended in the manual.
Oil level low in the sump	Check and fix the oil leakage.
Cooling fan not running	Replace fuse, cable and fan.
Oil filter choked	Replace the oil filter.
Oil quality deteriorated	Change the oil as recommended in the operations manual.

Overall, this air compressor troubleshooting web app streamlines the process of diagnosing and resolving issues, leading to increased efficiency, reduced costs, and improved equipment performance. However, it's essential to ensure that the app is user-friendly, regularly updated, and based on accurate and reliable information from reputable sources.

Literature Review

Daogang et al [6] the system model and real operating circumstances are defined; the goal is to reduce energy usage while manufacturing is underway. Discussion and analysis of the system's air compressor problems lead to the suggestion of concrete remedies. We propose the optimisation control of the retrofit scheme taking into account the actual power plant situation and compare the results of the model simulations simultaneously. Results show that the optimisation strategy proposed in this study can reduce operating costs for air compressor systems while increasing their energy-saving effects.

Yan et al [7] suggests a unique parameter update function to mitigate the shortcomings of the conventional PSO. Furthermore, the overall model of the huge system is formed by thoroughly considering the associations between the general numerous "sub-problems" based on the constructed subsystems. The air compressor's optimization control is built using this extensive model. The suggested model is tested in real-world scenarios and contrasted with the most recent models.

Eddouh et al [8] suggested the optimization problem for imperfect preventive maintenance (PM) by using minimal repairs and preventative replacements to enhance the air compressor's average availability under conditions of continual degradation while meeting minimal maintenance cost rate constraints. Every maintenance operation has a

price and a length, and the results of poor PM and corrective maintenance are based on the availability and cost of maintenance. Finding the ideal number of faulty repairs and replacement time is the goal of the suggested imperfect PM optimization model. Nonlinear programming is presented as a mathematical model, and a suggested solution is presented to address the optimization issue. All things considered, this article offers a thorough and formal study of the air compressor maintenance plan, providing insightful information about how to maximize faulty preventative maintenance.

Li et al [9] designed the differential evolution (DE) and the whale optimization algorithm (WOA) are combined in the suggested algorithm, DWOA. To identify the globally optimal solution, the suggested method can strike a compromise between exploration and exploitation. The original problem model is rewritten to be continuous by extending the compressor's flow rate range in a technique to addressing limitations that is also described. To show how well this strategy works, a case study is conducted. The continuous reformulated model is found to be easier to solve, and DWOA yields a good solution with a mere 1.61% deviation from theoretical values. Additionally, DWOA outperforms WOA, DE, and DE-WOA, another hybrid algorithm, in terms of accuracy and stability. As a result, this approach may encourage thorough compressor station optimization.

Ranganathan and Aruchamy [10] The suggested polymer material, polyphthalamide (PPA), reinforced with hybrid composites made of 5% PTFE and 30% glass fiber, was created by injection molding, and the results of the tests were used as input data for additional study. Solidworks was used to design the air intake valve body's bottom housing for this analysis, and Ansys Workbench 18.1 was used to take the generated model and use it for linked field analysis. An air compressor's four working conditions were tested for the FEA study by varying the materials' respective temperatures and pressures. In every operating situation, the output responses, such as total deformation and von-mises stress, are recorded.

System Environment

Table 2: Hardware Specification

Processor	Core i3
RAM Capacity	1GB
Hard Disk	160 GB
Mouse	Logical Optical Mouse
Keyboard	104 Keys
Monitor	16 inches
Mother Board	Intel
Speed	2GHZ
Floppy Disk Drive	100MB

Table 3: Software Specification

Front End	HTML, CSS, Java script
Environment	Visual studio code

Application Specification

Front End: Html, CSS, Java Script Html

In 1989, Tim Berners-Lee, Robert Caillou, and others were the first to invent HTML. Hyper Text Markup Language is what it stands for. To be hypertext, a document must have links that take the reader to different sections or even to other documents completely. The most recent release is called HTML 5. To govern the processing and presentation of text, computers communicate using Markup Languages. Tags and attributes are the tools that HTML utilises to accomplish this. A scripting language, like JavaScript, can be embedded into HTML, altering the content and behaviour of web pages. The style and arrangement of content are determined by the inclusion of CSS. As the current and former keepers of the CSS standards, the World Wide Web Consortium (W3C) has advocated for the use of CSS to enhance presentational HTML.

When creating content for display in a web browser, the de facto standard markup language is HTML, or Hyper Text Markup Language. Coding languages like JavaScript and Cascading Style Sheets (CSS) are common tools that help with this. In order to create multimedia web pages, web browsers accept HTML documents either from a web server or from local storage and then display them. Originally designed to provide visual clues, HTML now provides

a semantic description of a web page's structure. Components of an HTML page are called HTML elements. The produced page can have images and other objects, including interactive forms, inserted within it using HTML structures. HTML allows users to construct organised documents by designating the structural semantics of various text elements, including headers, paragraphs, lists, links, and quotes. Tags, which are written using angle brackets, demarcate HTML elements. Content is immediately introduced into the page with tags like <img /> and <input />. Surrounding and providing information about document text are other tags like and , which may also contain sub-element tags. While browsers do not show the HTML tags, they rely on them to understand the page's content.

Advantage of HTML

- Web pages development. HTML is heavily used for creating pages that are displayed on the world wide web.
- Web document Creation.
- Internet navigation.
- Cutting edge feature.
- Responsive images on web pages.
- Client-side storage.
- Offline capabilities usage.
- Data Entry support with HTML.

When creating content for display in a web browser, the de facto standard markup language is HTML, or Hyper Text Markup Language. Coding languages like JavaScript and Cascading Style Sheets (CSS) are common tools that help with this. In order to create multimedia web pages, web browsers accept HTML documents either from a web server or from local storage and then display them. Originally designed to provide visual clues, HTML now provides a semantic description of a web page's structure.

Components of an HTML page are called HTML elements. The produced page can have images and other objects, including interactive forms, inserted within it using HTML structures. HTML allows users to construct organised documents by designating the structural semantics of various text elements, including headers, paragraphs, lists, links, and quotes. Tags, which are written using angle brackets, demarcate HTML elements. Content is immediately introduced into the page with tags like <img /> and <input />. Surrounding and providing information about document text are other tags like and , which may also contain sub-element tags. While browsers do not show the HTML tags, they rely on them to understand the page's content.

Features of HTML

- User Friendly & Simple.
- Semantic Structure.
- SEO – Search Engine Optimization.
- Local Storage & Indexed DB – Client-side data storage.
- Offline Capabilities (PWA) with Cache API & Service Workers.
- Canvas for Game Development.
- Platform Independent.
- Media Support.

CSS

One way markup languages like HTML describe the presentation of a page is with style sheets, and one such language is Cascading Style Sheets (CSS). With HTML and JavaScript, CSS forms the backbone of the Internet. Layout, colours, and fonts are all considered part of the presentation, but using CSS, they can be easily separated from the information. Reduce complexity and repetition in the structural content; increase accessibility of content; allow multiple web pages to share formatting by specifying the relevant CSS in a separate.CSS file; and give more flexibility and control in the specification of presentation characteristics. By separating the information from the formatting, it is possible to create many versions of the same HTML page for different rendering ways. These methods include on-screen, print, voice (via a screen reader or speech-based browser), and Braille-based haptic devices. The World Wide Web Consortium (W3C) maintains the CSS specifications, which are eleven in number.

CSS's primary goal is to decouple presentational elements like layout, colours, and fonts from content.the third This division has many benefits, including making content easier to access, giving you more control over how presentation characteristics are defined, simplifying structural content by allowing multiple web pages to share

formatting in a single.css file, and improving page load speed for all of those pages by allowing the.css file to be cached. The ability to display the same HTML page in many styles for distinct rendering techniques, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices, is another benefit of separating formatting and content. If the information is viewed on a mobile device, CSS also includes rules for different formatting.

CSS Selector

CSS selectors into five categories:

- Simple selectors (select elements based on name, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo-class selectors (select elements based on a certain state)
- Pseudo-elements selectors (select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

With CSS, it's easy to publish material in many presentation forms using just a few minimal parameters. Among the numerous variables that make up nominal parameters are the user's location, the kind of device they are viewing the information on (desktop computer vs. mobile device), their stated preferences, and the various web browsers they are using. One way to define the style for many HTML components that are grouped together according to class, type, or relationships is via a stylesheet, which can be either internal or external. Doing it this way is far more efficient than inserting the same style information for every instance of the element. You can further reduce data transfer over a network by using an external stylesheet on many sites without having to reload it, since it is typically cached in the browser cache.

By making good use of inheritance and "cascading" in CSS, it is possible to style elements across the entire site from a single global style sheet. If you ever need to update or tweak the styling of the elements, you may do it by editing the rules in the global style sheet. Performing such upkeep required more effort, money, and time before CSS. Using a new style sheet for the same page is as easy as changing one line. This allows for the customisation of a page or site to various target devices and has benefits for accessibility. Even devices that can't interpret the style will still show the information. Web designers often resort to tactics like HTML tables when constructing pages without CSS, which might make them inaccessible to persons with visual impairments.

Features of CSS

- **Opportunity in Web designing:** Expertise in Cascading Style Sheets (CSS) and Markup Language (HTML) is required for entry-level web design positions.
- **Website Design:** There are a plethora of effects that may be achieved with CSS, including but not limited to: text colour, font style, paragraph spacing, column size and layout, background colour and images, layout design, display variants for different screens and device sizes, and many more.
- **Web Control:** Because of its control over HTML texts, CSS is simple to pick up and use. It works with both the Hypertext Markup Language (HTML) and its successor, XHTML.
- **Other Languages:** Other related technologies, like as Angular, PHP, and JavaScript, become more apparent after we grasp the fundamentals of CSS and HTML.

CSS Applications

Some critical applications of CSS are discussed and listed below:

- **Fast Page Loading:** We don’t need to mention the attributes of the HTML element every time if we use CSS. We need to specify one rule of CSS for an element and use it for every occurrence of that element. So, short code means high-speed download times.
- **Easy Maintenance:** We need to change the style if we want to make a global change. All web pages will have their elements automatically updated.
- **Superior HTML Styles:** Since HTML has less extended attribute arrays than CSS, we can give our HTML page a far nicer look using CSS.
- **Save Time:** We just need to define CSS once so we can use the same style sheet across all of our HTML pages. We have complete control over the style of each HTML tag and can apply it to an unlimited number of web pages.
- **Compatibility:** Content can be improved for one or more device types with the help of the cascading style sheet. Different mobile and desktop versions of the site could be made available for printing and other

portable devices.

- **Global Standards:** Attributes in HTML are currently being deprecated in favour of CSS. Therefore, to ensure compatibility with future browsers, it is recommended to start using CSS on all HTML pages.

Java Script

Along with HTML and CSS, the programming language JavaScript, sometimes abbreviated as JS, is a fundamental component of the World Wide Web. In 2022, nearly all websites (98%) will have included client-side JavaScript for page behaviour, with many also utilising third-party libraries. A specialised JavaScript engine is built into every major online browser, allowing users to execute code directly on their devices. JavaScript follows the ECMA Script standard and is a high-level language that is frequently just-in-time compiled.[10] It offers first-class functions, prototype-based object orientation, and dynamic typing. It is compatible with multiple paradigms and can handle imperative, functional, and event-driven programming. The Document Object Model (DOM), regular expressions, text, dates, and standard data structures are all accessible through its application programming interfaces (APIs).

Nothing related to input/output (I/O), including storage, graphics, or networking, is part of the ECMA Script standard. In actuality, the I/O JavaScript APIs are provided by the web browser or another runtime system. Although they were formerly exclusive to web browsers, JavaScript engines are now essential to many servers and apps. For this kind of application, Node.js is the go-to runtime system. Despite sharing a name, syntax, and standard library with one another, Java and JavaScript couldn't be more different in terms of design. JavaScript works with a lot of the C-style structured programming syntax. This includes things like while loops, switch statements, do while loops, and if statements. Scoping is an example of a partial exception; whereas var was the only keyword for function scoping in the original JavaScript, let and const were introduced for block scoping in ECMAScript 2015. Expressions and statements are treated differently in JavaScript, just as they are in C. Automatic semicolon insertion is one syntactic difference with C that lets you skip using semicolons to end statements.

Dynamic Typing

Like the majority of programming languages, JavaScript uses dynamic typing. Instead of an expression, a value is linked to a type. An example of this would be changing the initial binding of a variable from a number to a string. Duck typing is one of the methods for checking the type of objects in JavaScript.

Run Time Evaluation

JavaScript includes an eval function that can execute statements provided as strings at run-time.

Prototypes

JavaScript uses prototypes where many other object-oriented languages use classes for inheritance. It is possible to simulate many class-based features with prototypes in JavaScript.

Functions as Object Constructors

On top of their usual usage, functions also act as object constructors. By including the word "new" to the beginning of a function call, you can create a prototype instance that will inherit all of the constructor's methods and properties, including those of the Object prototype. To avoid immediately inheriting from the Object prototype, which might be set to null in previous contexts, ECMA Script 5 provides the Object. Create method. The internal prototype of a new object is defined by the object that is passed into the constructor as its prototype attribute. To add new methods, just change the prototype of the constructor function. You can also modify the prototypes of the built-in constructors in JavaScript, including Array and Object. Even though you can change the Object prototype, it's usually not a good idea. That's because most objects in JavaScript will take on the Object prototype's features and methods without realising it has been changed.

Functions as Methods

A function definition and a method definition are treated as one and the same, in contrast to many object-oriented languages. The difference, instead, happens when invoking the function itself: when an object's method invokes a function, the local this keyword of the function is tied to that object for the duration of the invocation.

Functional

A function is an object in JavaScript, and functions are first-class objects. This means that functions can have attributes and methods like call () and bind (). The definition of one function inside another is known as a nested function. In most cases, it updates the page with fresh data received from the server when the user interacts with it. With each call to the outer function, it is produced. Furthermore, every nested function creates a lexical closure. This

means that the inner function objects retain the lexical scope of the outer function, including all constants, local variables, and argument values, even after the outer function's execution ends. Anonymous functions are also supported by JavaScript.

Features of Java Script

- Scripting. JavaScript executes the client-side script in the browser.
- Interpreter. The browser interprets JavaScript code.
- Event Handling. Events are actions.
- Light Weight.
- Case Sensitive.
- Control Statements.
- Objects as first-class Citizens.
- Supports Functional Programming.

Applications of JavaScript

- Web Development.
- Web Applications.
- Presentations.
- Server Applications.
- Web Servers.
- Games

Visual Studio Code

A source-code editor developed by Microsoft using the Electron Framework, Visual Studio Code (or VS Code for short) is available for Windows, Linux, and macOS. Based on free source software. Is always on the go. Licence and privacy policies are accepted when you use Visual Studio Code. Debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git are some of the features. A user's preferences, keyboard shortcuts, theme, and the ability to install extensions are all customisable. Out of 71,010 respondents, 74.48% said they use Visual Studio Code, making it the most popular developer environment tool in the Stack Overflow 2022 Developer Survey.

Fortran, Go, Java, JavaScript, Node.js, Python, Rust, and C, C#, C++, Visual Studio Code is a source-code editor that works with many languages. Building Node.js web apps with the Blink layout engine is what the Electron framework is all about. Azure DevOps (formerly known as Visual Studio Online and Visual Studio Team Services) uses the same editor component (codenamed "Monaco") as Visual Studio Code. Basic support for the majority of popular programming languages is pre-installed in Visual Studio Code. Highlighting syntax, matching brackets, folding code, and customisable snippets are all part of this fundamental capability. Along with support for debugging Node.js, IntelliSense is built into Visual Studio Code for JavaScript, TypeScript, JSON, CSS, and HTML. You may add support for more languages to Visual Studio Code by downloading extensions from the Marketplace. These extensions are available for free.

You can customise Visual Studio Code by setting the current document's saving page, newline character, and programming language. This ensures its portability to any platform, language, and location. [words used to promote] You have the option to disable Visual Studio Code's data collection and transmission to Microsoft. The public has full visibility into the acquired data because the application is open-source. This includes the telemetry code. You can use Visual Studio Code as a free substitute for web development because it comes with numerous FTP extensions.

Reasons why visual studio is the best IDE for Web Development

- Visual Studio Code is Free
- Visual Studio Code Has a Large and Active Community
- Visual Studio Code Offers Stable Update
- Visual Studio Code Offers Intuitive Interface
- Quickly Installing Plugins in Visual Studio Code Is Easy
- Debugging with Visual Studio Code Is Reliable and Easy
- No Fuss with Themes in Visual Studio Code Material Theme

You may launch a genuine browser preview inside your editor and debug with Browser Preview for VS Code. Launching a headless Chromium instance in a separate process is how Browser Preview gets its power. Both Google Chrome and Microsoft Edge are acceptable choices here. Live server in Chrome or Microsoft Edge is used to execute VS Code programmes. How to Use This Easy Add-On to Have Visual Studio Code Live Server Automatically Refresh Your Browser. Of all the code editors available, Visual Studio Code is among the most widely used. You can make programming easier and more enjoyable with its free, streamlined interface and plethora of extensions. Any time you make a modification while using Live Server, the website will reload immediately. When using a Normie browser, however, you would need to refresh the website each time to see the changes. You can use the live server with Visual Studio Code for free because it is open-source software. By simply dragging and dropping an HTML file onto the live server, you may view the page in a new browser window as the server runs automatically. Servers, depending on their configuration, should be usable even when connected to the internet through their local area networks (LANs). Although clients on the local area network (LAN) can see it, the server could not function properly if it requires internet connection to carry out any of its responsibilities.

Visual Studio Code Features

- Develop
- Debug
- Test
- Collaborate
- Extend

Advantages of Visual Studio Code

- Cross-platform support
  - Windows
  - Linux
  - Mac
- Light-weight
  - Robust architecture
  - Intelli-sense
  - Freeware: Free of Cost- probably the best feature of all for all the programmers out there, even more for the organizations.
  - Many users will use it or might have used it for desktop applications only, but it also provides great tool support for Web Technologies like; HTML, CSS, JSON.

Results and Discussion

Input Design

Designing inputs that are orientated towards users is what input design is all about. Errors in data processing are most often caused by inaccurate input data. Simplifying the data is the primary objective of input design. A fully functional website is the proposed system. As an interactive design tool, it is powerful.

Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties:

- It should serve specific purpose effectively such as storing, recording, and retrieving the information.
- It ensures proper completion with accuracy.
- It should be easy to fill and straightforward.
- It should focus on user's attention, consistency, and simplicity.
- All these objectives are obtained using the knowledge of basic design principles regarding
- What are the inputs needed for the system?
- How end users respond to different elements of forms and screens?

Output Design

Each system's primary responsibility is the design of its output. Developers determine the required output types, think about the output controls, and create prototype report layouts during output design. Producers make and design printers' exterior outputs. Through the use of external outputs, the system is able to depart from recipients' actions to

either affirm or initiate them. A few of the outbound results are intended to be "turnaround outputs," meaning they can be used as a form and then inputted back into the system. Managers and end-users alike make use of the system's internal outputs. When it comes to reporting and decision making, they back up the management. The objectives of input design are:

- To develop output design that serves the intended purpose and eliminates the production of unwanted output.
- To develop the output design that meets the end user's requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the right person.
- To make the output available on time for making good decisions.

Fault Diagnostics

The fault diagnostics pages are used to detect the faults in the compressor using their symptoms. The second page consists of the product groups. The product link directs to its symptoms page. The symptoms can be confirmed by its confirmation points. Each symptom directs to its check points page and check points page in turn directs to the fault details page. The fault details page consists of the image reference, checking procedure, probable root cause, solution and the key points for the particular checkpoint.

Unit Converters

The unit converters page is a tool to easily convert various units between bar, kg/cm2 or m3/min, cfm and several other units. It contains separate calculators for volume, power, length, density, flow rate, pressure, temperature and weight.

Air System Calculators

Air system calculators page consists of air resistor sizing, ACFM calculators, Oil carry over, Pressure drop, ducting and energy saving.

Conclusion

As the technology emerges day by day this website will be more useful for the customers as well as the organization to see or show the products of the compressor. This web app is very efficient and saves time for both customers and organization. The customers can know more about the types of compressors in detail on the website. This will be very useful as the website contains all the details about the compressor. Compressors are used throughout the industries to provide shop or instrument air, to power air tools, paint sprayers, air conditioning, refrigeration etc. This online website will be more useful to pick the details about the compressor. Overall, this air compressor troubleshooting web app streamlines the process of diagnosing and resolving issues, leading to increased efficiency, reduced costs, and improved equipment performance. However, it's essential to ensure that the app is user-friendly, regularly updated, and based on accurate and reliable information from reputable sources. In conclusion, web apps offer numerous advantages that make them a popular and efficient choice for businesses and users alike. As technology continues to evolve, web apps are likely to remain a valuable asset for the digital landscape.

Reference

Li, Y., Pei, P., Ma, Z., Ren, P., & Huang, H. (2020). Analysis of air compression, progress of compressor and control for optimal energy efficiency in proton exchange membrane fuel cell. *Renewable and Sustainable Energy Reviews, 133,* 110304.

Nourin, F. N., Espindola, J., Selim, O. M., & Amano, R. S. (2022). Energy, exergy, and emission analysis on industrial air compressors. *Journal of Energy Resources Technology, 144(4),* 042104.

Nourin, F.N., Espindola, J., Selim, O.M., & Amano, R.S. (2022). Energy, Exergy, and Emission Analysis on Industrial Air Compressors. *Journal of Energy Resources Technology*, *144*(4), 1-14.

Pothula, N. R. (2020). Design and Fabrication of Pedal Operated Air Compressor. *International Journal of Renewable Energy and its Commercialization*, *6*(1), 21-29.

Galinato, J.R.M., Bastasa, J.W.G., Subido, K.A.M., Hacla, R.P., & Nicolas, R.D. Development of a Pedal Powered Air Compressor Generator.

Daogang, P., Haoran, W., Liqun, G., & Jun, Y. (2022). Optimization research of energy saving operation of air compressor system in coal-fired power plant. *In IEEE 4th International Conference on Electrical Engineering and Control Technologies (CEECT)*, 882-886.

Yan, W., Sun, Y., Li, N., Wang, L., & Lou, J. (2021). Energy-saving optimization control of air compressor based on particle swarm optimization. *In IEEE Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, 653-656.

Eddouh, Y., Daya, A., Elotmani, R., & Touache, A. (2023). Imperfect Maintenance Model for Optimizing Air Compressor Availability. *International Journal of Performability Engineering*, *19*(4), 263-272.

Li, X., Cui, T., Huang, K., & Ma, X. (2021). Optimization of load sharing for parallel compressors using a novel hybrid intelligent algorithm. *Energy Science & Engineering*, *9*(3), 330-342.

Ranganathan, S., & Aruchamy, S. (2020). *Design and Analysis of Air Compressor Intake Valve Body Casing by Metallic Material and Hybrid Polyamide Composite (No. 2020-28-0377).* SAE Technical Paper.