International Journal of Scientific Research in Computer Science, Engineering and Information Technology

ISSN : 2456-3307

OPEN ACCESS

Available Online at : www.ijsrcseit.com

doi : https://doi.org/10.32628/CSEIT243646

Dockerized Application with Web Interface

Abhishek M Nair[1], Sivaiswarya CK[2], Sidharth S[3], Visakh KK[4], Jibin Joy[5]

[1-4]B.Sc. CS Scholar, Department of Computer Science, Yuvakshetra Institute of Management Studies, Palakkad, Kerala, India

[5]Assistant Professor, Department of Computer Science, Yuvakshetra Institute of Management Studies, Palakkad, Kerala, India

ARTICLEINFO

Article History:

Accepted: 20 March 2024

Published: 06 April 2024

Publication Issue

Volume 10, Issue 2

March-April-2024

Page Number

412-419

ABSTRACT

Developing an application can be a task if any kind of conflict arises during deploying the code or while running them and it can be due to the storage or the code being not supported by the other party's system. Thus to provide a solution for this matter, we are introducing the project concept of Dockerized application deployment through a web interface. This proposed project combines the efficiency of Docker containers with a web interface to create a platform for running and managing applications easily. When a programmer or a developer or anyone in the field of programming has conflict in uploading, running or deploying their application code from another programmer's system to their own due to the inefficiency or lack of facilities in their system, they can use this web interface as a solution. Especially during the time of any rush, they can opt for this web interface as it does not require the installation of a local Docker software and any extra dependency management, as installation of Dockers are a bit time lagging. One of the main factors of this project is that this web interface can be run in any kind of computer system without any extra facilities being added to it. Whether the system is less efficient or high efficient regardless of the type of the system, this web interface is easy to access for the users. Users can upload their application code, build Docker images, and run them directly from the web interface. With the advantage of Docker's utility methodologies for shipping, testing and deploying code, you can reduce the delay between writing codes and running applications .It has additional features like users can define environment variables for their applications, configure network settings for container communication ,mount persistent volumes to store application data with help of virtual cloud, implement user roles and permissions for secure access control .The front end of the web page is created using NEXT Programming Language meanwhile the backend is applied using NEXT, Docker and Python Flask API. About NEXT Programming Language that in this language, when the front-end is applied the backend function gets directly deployed making us use less effort in creating the webpage. It's a newly created advanced programming language. Overall,

this Dockerized application deployment web-interface offers a user-friendly and efficient solution for developers, system administrators, and DevOps teams, streamlining the application development and deployment process.

Keywords : Docker, Shell Access, Static/File Hosting , Log Analyser

I. INTRODUCTION

The world grows on a constant innovation, technology changing from time to time by creating more progress in the society. Through technology various components have been introduced to humans for progress of this society and applications are the major part of this progress. They streamline tasks, unlock new possibilities for us, and empower individuals across various fields acknowledge their inner skills. However, traditional application development can face limitations in terms of portability, scalability, storage and easy accessing etc. thus as a solution for these factors, we are introducing our project to you. Our proposed project addresses these challenges by leveraging the combined power of Docker's containerisation technology, storage of virtual cloud and easy accessibility of web interfaces making the process of developing applications more efficient than what it was before.With the advantage of Docker's utility methodologies for shipping, testing and deploying code, you can reduce the delay between writing codes and running applications. By the combining of Docker and cloud accelerate development by enabling rapid deployments and streamlined testing processes. Faster development cycles, allowing teams to iterate and innovate at a much quicker pace. From the security provided through shell commands, confirming log data through log analyser, hosting files dynamically through static hosting, this interface have provided utilities for a local developer to develop their own application. This project acts as a technical support to developing teams, testing teams and other fields that work in application creation. As this introduction ignites, our project strives to empower individuals and contribute to the ever-evolving world of innovation by making application development and access more accessible and efficient. This, in turn, can foster the creation of new ideas and solutions that have the potential to positively impact the world around us.

II. NEED FOR STUDY

Our world is constantly changing so does technology. And we know that technology is created to reduce human efforts and to make works easier to implement. One of the main substance that bloomed from technology in today's world is applications. Applications are accessible anywhere and everywhere. But creating an application can be tiresome when sending it from developer to devOps team or to testing committee and the app code not being supported in their system due to lack of facility. To avoid this, software named Dockers were introduced to developer community. Still installing a Docker software into a system can be time consuming and might be having adapting issues. So to avoid these factors we have developed a web interface and have implemented Docker's containerisation functionalities in it with virtual cloud. Through this project, one can easily deploy their app codes, even deploy their finished application, do a log data analysis, host files dynamically and work on them.

III. LITERATURE SURVEY

The research contents that's being denoted below shows the usage of Dockers, Docker containers, containerization, connecting it with cloud for more effective working and its advantage towards the outside world of development. These contents have heavily influence on the project that's being developed by us and shows out a basic study regarding the contents that comes under the project topic, that is Dockerized Application with Web Interface

Software reuse errors resulting from differences in development environments are a common occurrence in software development, but Docker can successfully address these issues. Docker can create the same environment for software at any stage of development or testing. Stated differently, it modifies the container configuration to ensure that all configurations within it depend on each other consistently and to allow software development, testing, and release to all take place in the same environment. Docker offers three benefits. It can make use of the picture repository first. It can also do tests for continuous deployment. Thirdly, it utilizes resources at a high level. The role-based authority management system is implemented on the Docker platform in this study.[1]

Administrators and developers can benefit from a few features that Docker offers. With the help of the open platform Docker Engine, which is a lightweight, portable runtime and packaging tool, programs can be created, distributed, and executed. Additionally, it offers Docker Hub, a cloud-based application sharing service. By switching from standard virtual machines to docker containers, costs can be decreased. It significantly lowers the price of reconstructing the cloud development platform. When apps are put inside containers, Docker offers the ability to automate the process. Docker gives an additional layer of deployment engine to an environment using containers, where applications are virtualized and run. Docker is designed to provide an environment that is lightweight and speedy[2]

The advent of a novel procedure known as Containerization facilitated virtualization on the operating system level, while virtualization facilitates absorption at the hardware level. Host operating systems, which share pertinent libraries and resources, are used in containerization. Because there is no guest OS present, it is more efficient. The application-specific binaries and libraries can be executed on the host kernel, which speeds up execution. Applications and their dependencies are combined by the Docker platform to form the containers. These containers are constantly running in isolated space atop the kernel of the operating system. Docker's containerization capability ensures that the environment is compatible with any relevant applications[3]

The experiment involved three scenarios, with the first and last having two phases and the second having only one, in order to determine the differences between virtual machines and docker engines. First, the idle state resource usages of the CPU, RAM, and disk storage are observed and compared between the virtual machine and container in the first case. In scenario one, the last stage entails running a single instance and tracking response times at various intervals. The second scenario involves placing the single occurrence under stress and observing how the response times respond to the longer intervals under stress. In the fina l scenario, response times are measured in two phases: one under normal settings and the other under stressful ones. The results of this three-tier experiment indicate that, while virtual machines and containers maintained similar response times during the initial request interval, containers performed better in terms of resource usage while idle, putting less strain on the host computer. In terms of response times, they were likewise noticeably faster. Additionally, during stressful circumstances, when the virtu al machine

nearly quadrupled in sluggishness, this is most apparent.[4]

Docker streamlines application deployment by requiring an image to be built only once, which allows it to be deployed on any system that has the Docker daemon installed. A central platform allows for the sharing of images, making it simple to download and alter them to fit one's needs. Because an examiner can generate numerous Docker images based on the same root image but with different configurations to be benchmarked, Docker is a very appealing option for software benchmarking investigations. Additionally, investigations are simple to parallelize and repeat because images are portable and reusable. Lastly, there is also the option to restrict the amount of computational power resources available, such as the amount of RAM or CPU cores that each operating container uses.[5]
Through the use of technologies like Docker, containerization has accelerated the acceptance and implementation of Micro services Architecture, revolutionizing software development and deployment. The container, a small, lightweight, and portable device that packages an application along with all of its settings and libraries to provide consistency and flexibility across a variety of systems, is the central component of the approach. Due to their ability to provide a framework that allows services to be segregated and maintain adequate coordination while yet having individual functionality, containers are ideal for developing micro service architectures. The most well-known and simple tool for creating and executing containers is Docker, which makes it possible to build and test services across a variety of settings without any problems[6]

In the past, cloud computing service providers have offered a range of services to clients by utilizing virtualization, a method that creates virtual servers and computing capabilities on a single hardware set. Micro services are a more recent architectural style for developing cloud applications, which was made possible by extensive virtualization. A cloud application using micro services architecture is constructed as a collection of services. It is possible to individually build, test, and implement each service. One of the key technologies that makes Micro services architecture possible is containerization, and one of the most popular ways to do this is via Docker.[7]

Because containerization is lightweight and portable, it has become a popular method for cloud application deployment. The growing popularity of containers has made it clear that effective live migration strategies are required to provide resource optimization, fault tolerance, and workload balancing. When implementing containerized apps in the cloud, understanding the effects and expenses of container live migration is crucial. Comprehending the financial and performance ramifications of live migration can facilitate the formulation of optimization strategies and help make decisions that lead to productive and economical cloud deployments.[8]

Similar in behaviour to a lightweight virtual machine, Docker is a container virtualization technology that has arisen as a complement to virtual machine technology, albeit with a little less process isolation. They are simpler to share and lighter
.Using the primary operating system's kernel sharing mechanism, Docker container virtualization is a virtualization solution for executing many apps independently on the host. A common term for container virtualization is operat ing- system-level virtualization, which enables the operation of several programs on a single host. Additionally, it demonstrates how smaller Docker containers are when running programs and do not need a guest operating system. As a result, it ought to run on the server with lighter architectural advantages. [9]

In order to better balance the competing resource utilization requirements among isolated groups, the

Docker platform can separate the resources handled by a single operating system into isolated groups. The effective use, relocation, and expansio n of laboratory resources on cloud platforms are made possible with the use of distributed utilizations. Stud ying the application of computer cloud computing technology based on the Docker platform in laboratory management is therefore extremely valuable practically. Based on Linux container technology is Docker. In addition to challenging the IT system's service model, cloud computing also creatively challenges how computers and information networks are used. On the one hand, distributed computing's scalability and flexibility are further enhanced with the aid of cloud computing[10]

IV. SYSTEM REQUIREMENTS NEXT.JS

Next.js is an open source web development framework created by the private company Vercel providing React-based web applications with server-side rendering and static website generation. It extends React by providing additional features and functionalities, streamlining development workflows. It offers various ways to fetch data.Next.js allows generating static HTML pages at build time, offering superior performance for content that doesn't change frequently. Next.js provides a built -in file-based routing system, simplifying the mapping of URLs to components. Next.js allows creating API routes directly within your project, enabling the development of server-less functions for handling data access or other backend logic.Next.js is a
powerful and versatile framework that empowers developers to build modern, performant, and SEO-friendly React applications.

DOCKER

Docker is an open source platform that enables developers to build, deploy, run, update and manage containers- standardized, executable components that combine application source code with the operating system (OS) libraries and dependencies
required to run that code in any environment. Docker gives software developers a faster and more efficient way to build and test containerized portions of an overall software application. This lets developers in a team concurrently build multiple pieces of software. Each container contains all elements needed to build a software component and ensure it's built, tested and deployed smoothly. Docker enables portability for when these packaged containers are moved to different servers or environments.

PYTHON FLASK API

Flask is a lightweight Python framework widely used for building web applications and APIs (Application Programming Interfaces). It offers a flexible and minimalist approach compared to larger frameworks, making it ideal for projects ranging from simple prototypes to sophisticated web services. An API acts as an interface that allows applications to communicate with each other. It exposes specific functionalities of an application, enabling other applications to access and utilize its data or services. Flask's lightweight nature makes it easy to learn and use, allowing developers to quickly prototype and build APIs. Flask provides a core framework with minimal dependencies, allowing for customization and integration with various libraries and tools. While lightweight, Flask can handle complex APIs and scale well for larger applications. Flask is a powerful tool for building web APIs in Python. Its simplicity, flexibility, and scalability make it a popular choice for developers of all leve ls, empowering them to create dynamic and efficient web services

HTML/CSS

Hypertext mark-up language is a scripting language used for writing that are in web pages. It specifies layout and linking commands present in the hypertext document themselves the word hypertext refers to the nonlinear information on the document, which helps

to navigate through the pages. It is a simple language used to design and describe the layout of web page for shop HTML also supports multimedia and document links consists of special courts which is embedded in in text ads formatting .Cascading Style Sheets (CSS) is used to format the layout of a webpage. With CSS, you can control the colour, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colours are to be used, different displays for different devices and screen sizes, and much more. CSS is the language we use to style a HTML document CSS describes how HTML elements should be displayed.

V. METHODOLOGY

Accessing the project through a web interface that has a login at the start where the users can input their credentials to ac cess in. There'll be four functionalities presented to you in project web interface. Shell Access, Static Hosting, Log Analyser and Upload Docker Image. Shell Access is where you can conduct tasks like system management and problem diagnosis, among others, we can employ shell commands. Because the web interface in the project was made using the Linux shell, we may utilize it safely and securely with the aid of these shell command lines. The method that happening here is bash commanding/shell scripting. In Static hosting aka file hosting functionality you can store application data, host any kind of user-generated content (html/css or java files), or provide static files for the web interface, giving users access to a dynamic file system. You can access these files any time through this static hosting. Next functionality of the project is Log Analyser, this functionality helps in analysing your log data and help you find any threat in the data which can a affect your accessing. The analyser shows you the treat through a graphical format for basic understanding. You only need to upload your log data files and the system will generate the graph automatically. It also shows you what type pf threat your file has been effected with. Upload Docker Image as the fourth in which, you can configure and create a Docker image for your purpose with the same functionalities of an authentic local Docker software. This project mainly focusses on developers and the ones working in the development sectors in IT fields.

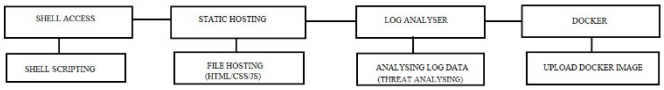

Figure 4.1 Basic Working of Dockerized Application With Web Interface

Login Page

Here the visual of login page, through this one can login into the system using their email and password.

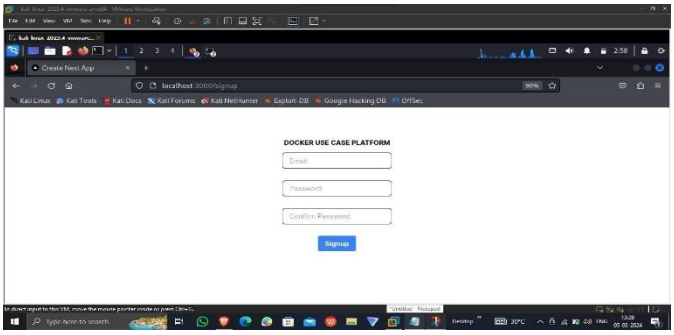

Fig 1.1

Shell Access

Here is a visual of this functionality, through shell accessing you can provide security through shell commands by shell scripting.

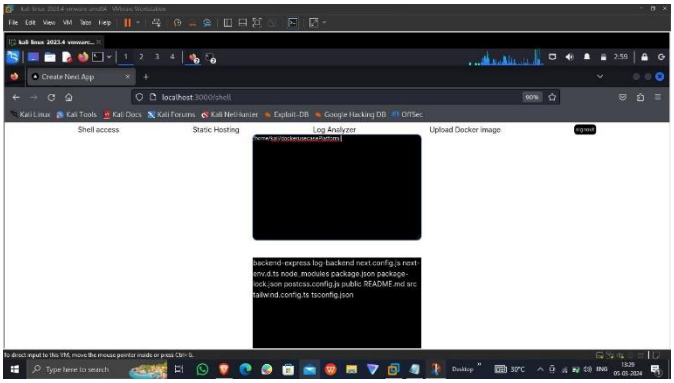

Fig 1.2

Static Hosting

Here is a visual of file hosting aka Static Hosting , you can host, store any kind of files here whether its an html or java or css file, its accessible dynamically.

Log Analyzer

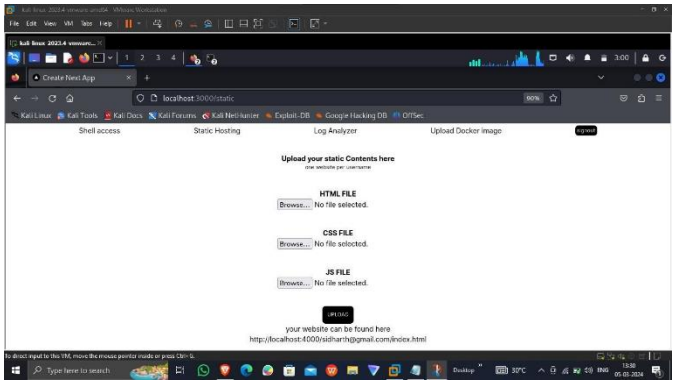

Fig 1.3

This functionality helps in analysing your log data and help you find any threat in the data which can a affect your accessing.

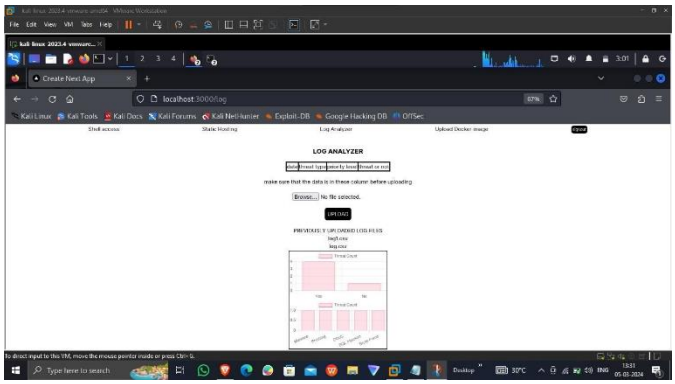

Fig 1.4

Upload Docker Image as the fourth in which, you can configure and create a Docker image for your purpose with the same functionalities of an authentic local Docker software. This project mainly focusses on developers and the ones working in the development sectors in IT fields.

VI. RESULT AND DISSCUSSION

The result regarding development and running of the project Dockerized Application With Web Interface, in short, has provided a productive and good-functioning setup for developers, programmers and other people who are keen on this particular field. The project has resulted in 90% of successful running. In order to ensure the program was suc cessfully deployed, the project underwent multiple stages of development including system analysis, design, implementation as well as testing. Every kind of system that exists ranging from very efficient ones to their less effective counterparts can access the project. The project have succeeded in all fields of requirements mentioned by the consumers, from shell accessing to file hosting etc. The project was implemented using Next.js and Python Flask, Docker features while HTML and CSS were used for its appearance have given us with a well done structure result of the project.

Regarding the future discussion of the project, after the successful deployment of it as per requirements mentioned now, into future we can implement more functionalities into the system from the current functionalities implemented in it. It is quite easy to integrate the system and all of its features. Because it guarantees confidentiality, speedy operation, and improved storage functionalities, the integrated version of this project can be employed in collaborative fields. Especially you configure the system as per the requirements of one cooperative company as per their needs then another way for another cooperative company as per their needs. In this, configure statement, it meant to add extra Docker images to deploy applications as per company developer's requirement. More functionalities in this project are log analyser, file hosting, these fields can be made more advanced into future. For the system's future development to be more efficiently developed, other modules can be added as needed. This project may be turned into web application software in the future, which could assist users in developing and implementing their ideas. Overall, this project can be developed more and more efficient as per time goes by.

VII. CONCLUSION

In to the conclusion, the "Dockerized Application With Web Interface" project has created a efficient, well working environment for the developers,

programmers and other users who are interested in this field. The project applicable to every single users out there. Throughout development of the project, it went through multiple stages including system analysing, designing, implementation, and testing, to ensure the successful deployment of the software. This project is accessible in any type of system, from highly efficient to least efficient. The project have been developed using the docker functionalities with Next.js and Python Flask meanwhile the designing concept went with HTML/CSS.The coding, sample testing reports, modules and other details regarding the project is mentioned in the report.

VIII. REFERENCES

[1]. China Li You and Hui Sun ' Research and Design Based of Docker Technology Based Authority Management System" Information Center, Hebei Vocational University of Industry and Technology, Shijiazhuang, China, Department of Automotive Engineering, Hebei Vocational University of Industry and Technology, Shijiazhuang. 13 April 2022 [1]

[2]. Babak Bashari Rad, Harrison John Bhatti, Mohammad Ahmadi "An Introduction to Docker and Analysis of its Performance" Asia Pacific University of Technology and Innovation Technology Park Malaysia, Kuala Lumpur, Malaysia, IJCSNS International Journal of Computer Science and Network Security, VOL.17 No.3, March 2017[2]

[3]. AmitM Potdar , Narayan DG , Shivaraj Kengond , Mohammed Moin Mulla "Performance Evaluation of Docker Container and Virtual Machine" Third International Conference on Computing and Network Communications (CoCoNet'19)[3]

[4]. Research By Mohammad Al Burhan., "Differences between Dockerized Containers and Virtual Machines A performance analysis for hosting web-applications in a virtualized environment" ,Dept. Computer Science & Engineering Blekinge Institute of Technology SE – 371 79 Karlskrona, Sweden. 18th May 2020[4]

[5]. Martin Grambow, Jonathan Hasenburg, Tobias Pfandzelter, David Bermbach., "Dockerization impacts in database performance benchmarking","Dockerization Impacts in Database Performance Benchmarking". Paper MCC.2018.1. TU Berlin & ECDF, Mobile Cloud Computing Research Group. 2018.[5]

[6]. Jon Mukaj., "CONTAINERIZATION: Revolutionizing Software Development and Deployment Through Microservices Architecture Using Doceker and Kubernetes", Research Paper Submitted To The Faculty Of Architecture and Engineering of Epoka University. July, 2023[6]

[7]. Jesse Dipor, Weidong Liao., "Cloud Computing and Docker Containerization: Research Paper, March 2019, Proceedings of the West Virginia Academy of Science 91(1)".[7]

[8]. Rachida ABOUNACER, Karim AFDEL, Amine BOUAOUDA., "Resource Utilization andCost Implications of Container Live Migration in Clouds:An Approach Performed on Amazon Web Services (AWS),Computer Systems and Vision Laboratory Department of Computer Science Faculty of Sciences, Ibn Zohr University Agadir, Morocco. September 1st 2023.[8]

[9]. Rendra Felani , Moh Noor Al Azam , Derry Pramono Adi , Agung Widodo , Agustinus Bimo Gumelar " Optimizing Virtual Resources Management using Docker on Cloud Applications" IJCCS (Indonesian Journal of Computing and Cybernetics Systems) Vol.14, No.3, July 2020.[9]

[10]. Research By Suiming Yang1, Hanshuang Jia, Yun Bai, "Research on the Application of Computer Cloud Computing Technology in Laboratory Management Based on Docker Platform", ZTE College, Xi'an Traffic Engineering Institute, Xi'an, Shaanxi 710300 2 School of Civil Engineering, Xi'an Traffic Engineering Institute, Xi'an, Shaanxi, 710300, 2023 (2021) 012016[10]