Abdullah Arif

EDUCATIONAL CHALLENGES OF CONTAINER TECHNOLOGY

Masters of Science thesis
Information Technology and
Communication Sciences
Examiner: Kari Systä
Terhi Kilamo
11/2023

ABSTRACT

Abdullah Arif: Educational Challenges of Container Technology
Master of Science Thesis, 54 pages
Tampere University
Master's Degree in Software, Web & Cloud
November 2023

This thesis conducts an investigation into the impact of containerization technology on educational programs and the professional job market. The objective is to explore the role of containerization in enriching students' understanding of software development methodologies, particularly Continuous Integration/Continuous Deployment (CI/CD) and DevOps practices. Additionally, the research assesses the implications of containerization on practical skill acquisition among students and examines the competencies sought by employers in individuals with containerization expertise. Challenges identified in educational frameworks for container technology are discussed, highlighting the need for adaptive educational programs. Survey results from students in the container technology landscape underscore these challenges. The findings emphasize the importance of educational programs evolving to meet the demands of the container technology domain. The research also addresses the rising demand for containerization expertise in the job market, informing job seekers about industry requirements. Beyond education and the job market, the study explores container technology's potential impact on software development and IT infrastructure. Overall, this thesis contributes to the discourse on containerization's role in education and the job market, providing insights for stakeholders and facilitating adaptation to container technology

Keywords: Containerization, Containers, Docker,

The originality of this thesis has been checked using the Turnitin Originality Check service.

PREFACE

I extend my heartfelt gratitude to Professor Kari Systä, whose invaluable guidance and unwavering support were instrumental throughout the thesis writing process. Your expertise and encouragement have been a constant source of inspiration.

I would also like to express my appreciation to my dear friends, Hamza Ahmed Malik, Arsalan Ahmed Siddique, and Muhammad Shahzaib, for their unwavering motivation and steadfast encouragement. Your belief in me propelled me forward.

Last but certainly not least, my deepest thanks goes to my family. Their unwavering support and understanding have been the cornerstone of my academic journey.

Tampere, 14 November 2023

Abdullah Arif

CONTENTS

1. INTRODUCTION 6

2. BACKGROUND 9

2.1 History 9

2.2 Popularity of Containers in the Software Industry 11

2.3 The Container Ecosystem 12

3. INTRODUCTION TO RESEARCH METHOD 14

3.1 Educational content Evaluation 14

3.1.1 Container topics in academic courses 15

3.1.2 Container topics in non-academic courses 18

3.2 Job analysis 24

3.2.1 Analysis of Job advertisements 25

3.3 Student Survey 27

3.4 Survey of Cloud technology Education Research Reports 28

3.4.1 How cloud technologies continues to transform education 29

3.4.2 Cloud technologies for STEM eduction 30

3.4.3 Features of distance learning of cloud technologies for the organization educational process in quarantine 31

3.4.4 Cloud Computing Adoption in Higher Education Institutions, A Systematic Review 32

3.4.5 Containers & Docker Emerging Roles & Future of Cloud Technology 32

4. RESULTS 34

4.1 Challenges with respect to educational programs and Job market 34

4.2 Learning Challenges in Container Technology Survey 37

5. DISCUSSION 43

6. CONCLUSIONS 47

REFERENCES 49

LIST OF FIGURES

Figure 2.1 Docker Container 10
Figure 4.1 Respondents division based on familiarity of containers 38
Figure 4.2 Respondents division based on containers education................................ 38
Figure 4.3 Respondents division based on using container at workplace.................... 39
Figure 4.4 Educational Program & Professional Readiness 39
Figure 4.5 Respondents division based on pace of learning containers in educational programs.. 40
Figure 4.6 Respondents division based on opportunity to use containers in real world 40

LIST OF SYMBOLS AND ABBREVIATIONS

AWS	Amazon web services.
API	Application Programming Interface
BDD	Behaviour-driven development
CI	Continuous integration
CD	Continuous development
CLI	Command line interface
CPU	Central Processing Unit
CaaS	Container-as-a-Service
CTE	Career and technical education
EMC	Electromagnetic Compatibility
ECR	Elastic Container Registry
ELK Stack	Elastic search, Log stash, Kibana Stack
FreeBSD	Free Berkeley Software Distribution
GCP	Google Cloud Platform
HEI	Higher education institutions
IAAS	Infrastructure as a Service
ICT	Information and Communication Technology
IaC	Infrastructure as Code
IBM	International Business Machines Corporation
IT	Information Technology
LXC	Linux Containers
LXD	Learning experience design
MLOps	Machine learning operations
OS	Operating Systems
OCP	Open Container Project
OCI	Oracle Cloud Infrastructure
SQL	Structured Query Language
TDD	Test Driven Development
VMs	Virtual Machines
YAML	Yet another Mark-up Language

1. INTRODUCTION

In the fields of software development and deployment, containerization technology has been a disruptive force in recent years. Containers provide a lightweight, portable, and isolated environment for packaging applications and their dependencies [1]. This innovation has altered the way software is developed, tested, and deployed across diverse computing environment.

The fundamental concept behind containers lies in their ability to encapsulate an application along with its dependencies, ensuring consistent execution across different systems. This portability eliminates the notorious "it works on my machine" dilemma, allowing developers to create applications that run reliably in various contexts, from local development environments to production servers. Furthermore, containers offer a level of isolation that shields applications from conflicts and interference with other software components. This isolation not only enhances security but also streamlines troubleshooting and maintenance processes [2]. As a result, organizations across industries have embraced containerization, recognizing its potential to accelerate development cycles, improve resource utilization, and enhance overall operational efficiency.

This thesis aims to thoroughly explore the role of containerization in educational programs and the professional job market. It focuses on the practical skills student’s gain, such as creating and managing containers and orchestrating containerized applications. Additionally, it investigates how containerization enhances understanding of key software development methodologies like CI/CD and DevOps. The research also analyses the job market to identify career opportunities and sought-after skills in containerization. Ultimately, this thesis seeks to be a comprehensive resource for educators, students, and professionals, shedding light on the varied impact of containerization. Through systematic research, it aims to offer a thorough understanding of containerization's influence in education and the job market, contributing to discussions about its role in software development and IT infrastructure. We investigate the following research questions to explore:

RQ0: What is the main objective of this thesis regarding the effects of containerization on educational systems and job prospects?

RQ1: In what ways does the thesis aim to bridge the gap between theoretical understanding and practical skills related to containerization?

RQ2: How do student surveys contribute to understanding the practical implications of containerization on educational methodologies and real-world skills acquisition?

RQ3: What contribution does the thesis intend to make to academic discussions about the role and future prospects of containerization in software engineering and IT infrastructure?

Multi Vocal Literature review, A multi-faceted research approach will be employed to comprehensively examine the role of containerization in educational programs and the professional job market. A multi-faceted research mean's the research approach that is been utilized to thoroughly investigate the role of containerization in both educational programs and the professional job market. The first step involves an extensive Multi Vocal Literature review, which will serve as the foundation for understanding the existing body of knowledge on containerization, especially within the domains of education and employment. This review will identify key themes and gaps, providing context for subsequent investigations.

a. Educational content evaluation delves into the content and structure of containerization courses within educational programs. This step aims to dissect the practical skills that students acquire, ranging from container creation and management to the orchestration of containerized applications, and to ascertain how these skills align with industry demands.
b. A job market analysis. This step will involve the collection and analysis of data from job postings and employer requirements in the IT and software development sectors. The objective is to identify trends and competencies sought by employers, thereby bridging the gap between educational offerings and industry demands.
c. Student surveys will be conducted, these surveys will capture first-hand experiences, shedding light on how containerization contributes to students' understanding of crucial software development methodologies such as CI/CD and DevOps.

In essence, containerization technology has revolutionized traditional methods in software development and deployment. It offers a more efficient approach by packaging applications in lightweight, portable, and isolated containers. This ensures consistent performance across different computing environments, leading to enhanced operational efficiency and strengthened security measures.

The goal of this thesis is to thoroughly explore the impact of containerization on educational frameworks and professional employment opportunities. Through a methodically rigorous process involving extensive literature review, critical assessment of educational curricula, analysis of job market trends, and empirical investigation via student surveys, this study aims to provide a comprehensive understanding of how containerization influences educational practices and industry demands. By bridging the gap between theory and practical skills, this research aims to contribute valuable insights to academic discussions concerning the role and future prospects of containerization in software engineering and IT infrastructure.

2. BACKGROUND

The utilization of containers has radically altered how software is built, deployed, and managed. They offer a portable and compact way to package applications and their dependencies, allowing them to run consistently across different computing environments. Containers have gained immense popularity in the software industry due to their numerous advantages, such as scalability, efficiency, and flexibility.

2.1 History

The concept of containers has a long history that predates their recent surge in popularity. Here's a brief overview of their evolution.

The roots of container technology can be traced back to the UNIX operating system, which introduced the concept of "chroot" in the 1970s. Chroot allowed processes to run in isolated environments with their own file systems [3]. However, it had limitations and wasn't a true containerization solution.

In the early 2000s, projects like FreeBSD Jails and Solaris Zones introduced more advanced containerization techniques. These technologies allowed for better isolation of processes and resources, laying the groundwork for modern containerization [4].

The container technology we know today was largely popularized by Docker, a company founded in 2010. Docker introduced a user-friendly platform for building, packaging, and distributing containers. Docker containers encapsulated not only the application code but also all its dependencies, facilitating the constant deployment of software applications across different environments [5].

The technique of containerization used by Docker made it accessible to a wide audience of developers, and it quickly gained widespread adoption in the software industry.

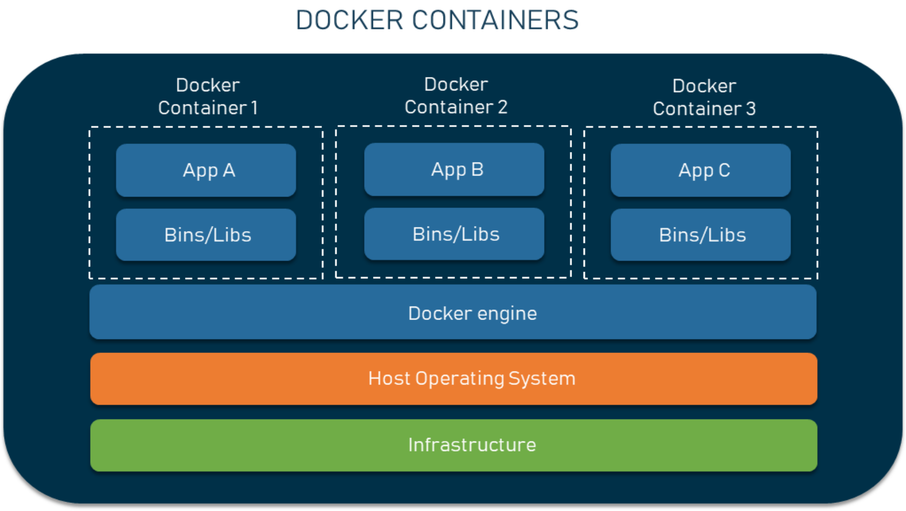

Figure 2.1 Docker Container

In the above image [6], Docker containers are represented by one or more rectangular boxes. A particular application or service, together with all of its dependencies and runtime environment, are contained within each container. Isolated instances that operate reliably in many environments are known as containers. The Docker engine is represented by a larger rectangle. Container orchestration and management is handled by the Docker engine. There are two primary parts to it, Daemon Docker. This is the host system's Docker container management background service. Command Line Interface (CLI) for Docker. This gives users the ability to communicate with the Docker engine via a command-line interface. Underneath the Docker engine box, the host operating system is modelled as a platform upon which the Docker engine operates. It could be Windows, Linux, or another operating system that works well. The Docker engine and containers are hosted on real or virtual servers, which make up the infrastructure layer. This could be hosted on-site, in the cloud, or a hybrid of the two. The host operating system is connected to the servers.

Kubernetes, on the other hand, was initially developed by Google and open-sourced in 2014. It deals with the problem of scalable managing containerized applications. It was created as a response to the practical difficulties brought on by the rising use of containers [7]. Large and complicated containerized systems can be easily managed by utilizing Kubernetes automation of container deployment, scaling, and orchestration. Since then, it has emerged as the industry benchmark for container orchestration and is essential to the containerization ecosystem.

The ability of Kubernetes to separate containers from the underlying infrastructure was one of its major innovations. The ability to define the desired state of their applications (such as how many replicas of a container should run) without having to worry about the specifics of where or how these containers would run was made possible by this decoupling [8]. Due to this conceptualization, Kubernetes developed into infrastructure-independent and could be used with a variety of container runtimes, including Docker.

As containers gained popularity, the need for orchestrating and managing them at scale became evident. Platforms for container orchestration such as Kubernetes emerged to deploy automatically, scaling, and management of containerized applications. The Google-developed Kubernetes platform has taken over as the industry norm for container orchestration and is widely used in production environments.

2.2 Popularity of Containers in the Software Industry

Containers have seen remarkable adoption in the software industry, driven by six key factors

1. Portability

Containers encapsulate applications and their dependencies into a single package, ensuring consistent behaviour across different environments. This portability allows developers to build applications once and run them anywhere, from development laptops to production servers and cloud platforms [9].

2. Efficiency

Containers are lightweight compared to traditional virtual machines (VMs) because they share the host operating system's kernel. This efficiency means that more containers can run on the same hardware, leading to better resource utilization and cost savings [10].

3. Rapid Development and Deployment

Containers enable a "build once, run anywhere" paradigm. Developers can create containers locally, test them, and then deploy the exact same container image to production. This reduces the "it works on my machine" problem and streamlines the development-to-production pipeline.

4. Isolation

Containers provide process and resource isolation, ensuring that applications do not interfere with each other. This isolation is essential for building secure and reliable systems [11].

5. Ecosystem and Tooling

Containers have a vibrant ecosystem of supporting tools and services. Docker, Kubernetes, and container registries like Docker Hub and Google Container Registry have become integral parts of the container ecosystem. These tools make it easier to build, manage, and scale containerized applications [12].

6. Cloud-Native Adoption

The rise of cloud-native architectures and micro services has further fuelled the adoption of containers. Containers are a natural fit for these modern application paradigms, allowing for easy scaling and resilience.

2.3 The Container Ecosystem

The container ecosystem is a rich and diverse landscape comprising various components, tools, and services that support containerization and container orchestration [13]. Let's explore some of the key elements of this ecosystem.

1. Container Runtimes

Container runtimes are responsible for executing containerized applications. Docker was the dominant container runtime for many years, but alternatives like container and rkt have gained popularity. These runtimes interface with the host operating system's kernel to provide isolation and resource management [14].

2. Container Registries

Container registries are repositories where container images are stored and distributed. Amazon Elastic Container Registry (ECR), Google Container Registry, and Docker Hub are popular examples. Developers can pull container images from these registries to run applications or push their own images for distribution [15].

3. Container Orchestration

Container orchestration platforms like container deployment, scaling, and management are automated using Apache Mesos, Kubernetes, and Docker Swarm. Kubernetes, with its robust features and large community, has emerged as the leader in this space [16].

4. Container Networking

Container networking solutions allow containers to communicate with each other and the outside world. Container orchestrators often provide built-in networking capabilities, but solutions like Calico, Flannel, and Weave offer additional networking features and flexibility [17].

5. Container Monitoring and Logging

Monitoring and logging are crucial for understanding the behaviour of containerized applications. For container monitoring and logging, applications like ELK Stack (Elastic search, Log stash, Kibana), Grafana, Prometheus, and Fluentd are frequently used [18].

6. Security and Compliance

Container security is a paramount concern. Tools like Clair, Aqua, and Twistlock help scan container images for vulnerabilities. Container security policies, network segmentation, and best practices like the Principle of Least Privilege are also essential for securing containerized environments [19].

7. Continuous Integration/Continuous Deployment (CI/CD)

CI/CD pipelines have become integral to containerized application development. Jenkins, Git Lab CI/CD, and Travis CI are popular CI/CD tools that integrate seamlessly with container ecosystems, enabling automated testing and deployment of containerized applications [20].

8. Serverless Computing

Azure Functions, AWS Lambda, and Google's Cloud Functions are examples of server less platforms. Also utilize containers behind the scenes. They facilitate programmers to react to events by running code without having to deal with the supporting infrastructure.

3. INTRODUCTION TO RESEARCH METHOD

In Chapter 3 of thesis, after conducted a multi-vocal literature review as a research methodology. To get a thorough understanding of the topic, this method entailed analysing and examining a wide variety of academic sources and research publications. In order to ensure that the examination of the literature is impartial and well-rounded and does not favour any particular point of view, a variety of ideas and viewpoints were included. The benefits and drawbacks of many studies were objectively assess thanks to the multi-vocal literature review method, which helped gain a more complete awareness of the research environment in this area.

3.1 Educational content Evaluation

Based on comprehensive analysis of a several of online courses on container technology, the following inquires have emerged in light of the information and resources they offer. The courses are based on the levels, e.g., beginner-level courses are tailored for individuals with little to no prior experience in the subject matter, focusing on foundational concepts and basic skills. In contrast, intermediate-level courses addresses to learners who already possess some familiarity with the topic and delve deeper into more complex concepts and practices. These courses assume a certain level of prior knowledge and often require learners to apply critical thinking and problem-solving skills to advance their understanding. To effectively select courses for analysis, several steps are followed. Firstly, criteria are established to define what constitutes a suitable course, with factors such as high ratings, popularity among learners, and relevance to container technology being key considerations. Next, various online learning platforms are explored, including Coursera and Udemy, CloudAcademy to identify potential courses. Within these platforms, lists of courses related to container technology are examined. Ratings and reviews from past learners are carefully reviewed to gauge the quality and effectiveness of each course. Additionally, the number of enrolments is taken into account, as higher enrolment numbers may indicate greater credibility and appeal. The reputation and trustworthiness of the organization offering the course are also assessed. Most importantly, the educational content of each course is scrutinized to ensure it aligns with the focus on container technology. Once all criteria are considered, courses that meet the requirements are selected for further analysis, with detailed notes kept to justify each choice and its relevance to the research objectives.

3.1.1 Container topics in academic courses

The course "Cloud Technologies and Systems" offered by the Faculty of Information Technology and Communication Sciences at Tampere University under the Computing Sciences Studies presents a comprehensive exploration of advanced studies in cloud technology. Positioned within the Faculty, this course is designed to equip students with an in-depth understanding of cloud platforms, particularly at the Infrastructure as a Service (IaaS) level. The curriculum delves into the foundational principles of virtualization, elucidating the mechanics and functionalities of this pivotal technology. Preceding knowledge expectations encompass a grasp of various virtualization types, alongside a conceptual understanding of cloud.

The learning objectives of the course are multifaceted, aiming to instil proficiency in the underlying principles of cloud platforms, with a specific emphasis on IaaS. Additionally, students are expected to acquire hands-on skills in working with Docker containers. The learning promises extend to covering the basics of different virtualization types, elucidating the definition and background of cloud based on standards, and practical engagement with Docker containers. Complementary knowledge areas include a deep dive into pricing models and optimization, as well as an exploration of security and privacy issues in accordance with relevant regulations. The course also covers the installation, management, and monitoring of IaaS systems. Specialist knowledge is a focal point, encompassing detailed insights into various hypervisor and virtualization techniques, software-defined networks, software-defined storage, cloud storage, and the hardware architecture of modern data centres.

In essence, the course is structured to provide a comprehensive and specialized knowledge base in cloud technologies, ensuring that students not only grasp the theoretical underpinnings but also gain practical expertise in deploying and managing cloud systems effectively. The diverse range of topics covered, from virtualization principles to specialized aspects of hardware architecture, positions students to navigate the complexities of contemporary cloud technology with both depth and breadth of understanding [26].

The "DevOps with Docker" course is designed for students at University of Helsinki is an intermediate level, meaning you should have some basic familiarity with Linux operating systems and web development, though it's not required. Throughout the course, you'll dive into the world of containerization using Docker, learning about important concepts like images and volumes. By the end of the course, students will be able to do quite a few things: they will know how to run applications inside containers, which are like lightweight, portable virtual environments, containerize existing applications, meaning they'll package them up to run in these containers; students will understand how to use volumes to store data outside of the containers, which is handy for things like databases or file storage; they'll also figure out how to map ports so that you can access your containerized applications over the internet; and finally, they'll discover how to share your own custom containers with others. The course wraps up with a simple pass/fail grading system, based on completing exercises throughout. By the end, you'll have practical skills and knowledge to apply DevOps practices confidently in your work.

The "DevOps with Docker" course equips intermediate-level students with practical skills in containerization using Docker. From running applications in containers to sharing custom containers, students learn essential DevOps practices. With a focus on hands-on exercises and real-world applications, the course empowers learners to streamline development and deployment processes efficiently [27].

The advanced studies course titled "Continuous Development and Deployment - DevOps" is designed to give students a deep dive into DevOps practices, emphasizing a holistic approach that spans technical, managerial, and business dimensions. Geared towards individuals with a solid foundation in information technology, the course mandates a prerequisite of good IT skills, ensuring that enrolled students possess a baseline proficiency for the advanced subject matter.

The overarching objective of the course is to equip students with a profound understanding of various virtualization techniques, placing a specific emphasis on the practical aspects of building and setting up Docker containers from a software development standpoint. The course extends to elucidate the principles of Continuous Delivery and Deployment, encompassing not only the technical intricacies but also the managerial and business considerations inherent in these processes. Additionally, students delve into the core concept of DevOps, exploring its underlying motivations and the role it plays in contemporary software development projects. Complementary knowledge areas are integrated into the curriculum, covering the construction and optimization of continuous deployment pipelines. The course also addresses the landscape of well-known cloud platforms and commercial options, providing a comprehensive view from both software development and business perspectives. Students gain insights into the software architecture implications of cloud environments and DevOps, along with an exploration of cloud-native architectures. Specialist knowledge domains form a significant component of the course, empowering students to apply DevOps methodologies in diverse organizational and business contexts. The course guides students in selecting the most fitting architecture for varied applications and business cases. Moreover, the course imparts the knowledge and skills necessary for implementing scalability and elasticity in cloud environments, reinforcing the practical applications of DevOps principles.

The learning promises of the course revolve around immersing students in the multifaceted aspects of DevOps software engineering. From business and organizational considerations to hands-on practices such as Continuous Deployment pipeline, automation, deployment, scaling, and monitoring of cloud applications, students are exposed to a comprehensive learning experience. Both theoretical background and practical applications are seamlessly integrated into the curriculum, ensuring that graduates emerge with a robust skill set and a nuanced understanding of DevOps practices in contemporary software engineering contexts [28].

“Cloud Platforms” is yet another intermediate studies course that serves as a pivotal exploration of contemporary cloud computing, catering to individuals with a foundational background in information technology. With an emphasis on practical applications and theoretical foundations, the course mandates a prerequisite of good IT skills and a basic knowledge of command line tools, ensuring that enrolled students possess the essential groundwork for engaging with the intermediate-level content.

The overarching objective of the course is to impart a comprehensive understanding of cloud platforms, encompassing key components such as the basics of virtualization, the utilization of commercial platforms, and the installation and management of open source cloud platforms. Students explore the detailed risks involved in virtualization, gaining a better understanding of the basic concepts that form the foundation of cloud computing. Complementary knowledge areas are seamlessly integrated into the curriculum, covering software-defined networks, software-defined storage, and cloud storage. The course extends to optimizing usage based on pricing models, administering and tuning the platform, and understanding the risks associated with commercial platforms. Moreover, students explore the cloud software stack, gaining a holistic view of the various elements that constitute modern cloud platforms. Specialist knowledge domains are a focal point, enabling students to delve into the internals of hypervisors and different virtualization techniques. The course addresses security considerations in commercial platforms, providing students with the skills to install and manage additional services on open source cloud platforms. Furthermore, students gain insights into the hardware architecture of a modern data centre, ensuring a well-rounded understanding of the underlying infrastructure that supports cloud computing.

The prerequisites set the stage for a robust learning experience, and the learning promises of the course align with a practical and hands-on approach. Students focus on installing, managing, and using modern cloud platforms, with a specific emphasis on commercial platforms such as Amazon Web Services, Microsoft Azure, and Google Cloud Platform. The curriculum extends to empower students to set up their own cloud platform, ensuring a well-rounded and applied understanding of cloud computing in contemporary IT landscape [29].

The suite of courses examined offers a comprehensive exploration through the intricacies of contemporary information technology, with a particular focus on cloud computing and DevOps practices. Spanning various difficulty levels—ranging from intermediate to advanced studies—these courses cater to individuals with diverse levels of expertise. The foundational course, "Cloud Technologies," sets the stage with an exploration of cloud service models and virtualization techniques, ensuring a solid grasp of essential concepts. Moving to the intermediate level, "Cloud Platforms" extends the learning by delving into commercial and open-source platforms, optimizing usage based on pricing models, and addressing hardware architecture. The advanced-level course, "Continuous Development and Deployment - DevOps," immerses students in the holistic realm of DevOps, covering virtualization techniques, continuous deployment pipelines, and the application of DevOps in varied contexts. The pinnacle of advanced studies, "Cloud Technologies and Systems," encompasses a broad spectrum, ranging from underlying principles in cloud platforms to specialist knowledge in hypervisors, software-defined networks, and modern data centre architectures. Collectively, this suite of courses forms a robust educational pathway, equipping students with theoretical foundations, practical skills, and specialized insights essential for navigating the dynamic landscape of cloud computing and DevOps practices.

3.1.2 Container topics in non-academic courses

The course "Introduction to Containers" was chosen for its relevance to the research topic on educational challenges of container technology. This beginner-level course provides a foundational understanding of containers, which is essential for exploring the associated educational hurdles. Its accessibility makes it suitable for a diverse audience, including product managers, junior developers, and individuals interested in learning about container technology. The course's high rating of 4.7/5 on Cloud Academy, along with its substantial enrolment of 19,974 students, attests to its credibility and popularity among learners. Insights gleaned from analysing this course will contribute to understanding fundamental concepts and terminology related to container technology, thereby informing discussions on educational challenges in the thesis.

“Introduction to Containers”, tailored to beginners, is designed to introduce the fundamentals of container technology. It assumes a basic level of development literacy from participants and recommends familiarity with concepts like servers, virtual machines, and Linux distributions, although such knowledge is not a mandatory prerequisite. The course's core objectives are to provide a comprehensive understanding of containers, encompassing their foundational principles and operational mechanisms. Additionally, it aims to facilitate comparative analysis, allowing participants to distinguish containers from similar technologies, and thereby appreciate their unique advantages. Exploring the various use cases of containers is another central objective, enabling learners to make informed decisions about their application. Importantly, the course promises to equip participants with practical skills for deploying containers effectively in real-world contexts, with a particular emphasis on the process of containerizing applications. The learning commitments in this course entail the development of a profound grasp of container technology, encompassing core concepts and functional aspects. Through comparative assessments, participants will differentiate containers from related technologies, fostering a clear understanding of their specific benefits. The course will also guide learners in identifying and delving into various use cases for containers, preparing them to make informed decisions regarding their implementation. A significant emphasis is placed on hands-on expertise in the deployment and utilization of containers, with a focus on the practical process of containerizing applications in real-world scenarios [21].

The "Introduction to Containers" course was picked because it's a beginner-friendly way to learn about containers, Docker, Kubernetes, and Cloud services like AWS. It's for anyone who wants to understand these concepts. The course is well-liked, with a rating of 4.4/5 on Udemy, and it's been taken by over 20,000 students. By looking at this course, we can better understand the basics of container technology and how it's taught, which will help us talk about educational challenges in the thesis.

“Introduction to Containers” from online platform Udemy is tailored for beginners with no prerequisite knowledge required. It is designed to provide a foundational understanding of container technology, specifically focusing on Docker and Kubernetes. The course's central objectives include helping learners grasp key concepts related to containers, comprehend the functions and applications of Docker and Kubernetes, and explore cloud services for container deployment, particularly on platforms like AWS. The learning promises associated with this course include gaining a comprehensive understanding of the possibilities offered by containers and their basic concepts. Participants will delve into Docker and Kubernetes, discovering how these technologies can work in tandem. The course covers fundamental aspects of containers, including container hosts, repositories, and images. Moreover, it delves into container service options provided by leading cloud providers such as AWS, Azure, and Google Cloud, offering insights into the practical implementation of containers within these environments [22].

The "Introduction to Containers w/ Docker, Kubernetes & OpenShift" course is an intermediate-level course that teaches how to use containers to move applications between different environments quickly. It covers building cloud-native applications with Docker and Kubernetes, as well as understanding Kubernetes architecture for managing containers. With a high rating of 4.4/5 on Coursera, based on 706 reviews where 91% are positive, and over 82,000 students enrolled, it's a popular and trustworthy course. Analysing this course will help us understand more about intermediate container concepts and how they're taught, which is important for discussing educational challenges in the thesis.

A course named "Introduction to Containers with Docker, Kubernetes & Open Shift" on Coursera is an intermediate-level course is designed to provide participants with a comprehensive understanding of contemporary containerization tools and technologies, including Docker, container registries, Kubernetes, Red Hat, Open Shift. It assumes no specific prerequisite knowledge and focuses on teaching learners how to build cloud-native applications using these tools and technologies. The primary objective of this course is to equip participants with the knowledge and skills to develop, deploy, and scale cloud-native applications in various cloud environments, be it public, private, or hybrid. Learners can expect to delve into a wide array of topics, including Docker objects, Docker file commands, container image naming, networking, storage, plugins, and Kubernetes CLI usage for managing workloads and applying basic commands. The course also covers advanced Kubernetes concepts like Replica Sets, auto scaling, rolling updates, Config Maps, Secrets, and service bindings. An additional highlight of the course is a comparative study of the similarities and differences between Open Shift and Kubernetes. The learning promises of the course revolve around the practical application of container technology to enable the swift movement of applications across different environments. Participants will gain the ability to build cloud-native applications using Docker, Kubernetes, Open Shift, and Istio. Moreover, they will acquire the skills to describe and leverage the Kubernetes architecture for setting up and operating a comprehensive container management system across the application lifecycle. The course also imparts the knowledge and expertise required to create and utilize YAML deployment files for the configuration and creation of essential resources, such as pods, services, replica sets, and others, in a declarative manner [23].

The "Basics of Using Containers in Production" course is aimed at intermediate learners and covers essential topics related to container usage in a production environment, including designing micro services and 12-factor apps. It helps you understand various options for running containers and addresses common security concerns. Additionally, it highlights the importance of container orchestration and discusses which types of applications are suitable for containerization. The course also provides insights into logging and monitoring practices specific to containers. With a solid rating of 4.4/5 on the Cloud Academy website and over 11,000 students enrolled, it's a popular choice among learners. Analysing this course will provide valuable insights into intermediate-level container usage practices, contributing to discussions on educational challenges within the thesis.

"Basics of Using Containers in Production" is yet another intermediate-level course which is tailored for participants with a foundational understanding of containers and an appreciation of their utility, but who seek a deeper understanding of container technology and its practical applications. Basic technical literacy and a grasp of cloud application architecture are assumed prerequisites. The course's primary objectives are to provide learners with a comprehensive understanding of fundamental container concepts. Participants will explore various container systems and gain insights into container orchestration. The course emphasizes the importance of building and running 12-factor apps on containers while addressing container security concerns. The twelve-factor app methodology uses straightforward setup instructions to speed up the on boarding of newly hired developers. It also guarantees that your programme can be simply deployed on cloud platforms and functions flawlessly across different systems. It also makes it simpler to accommodate additional users and update your software more frequently without requiring significant adjustments[24]. Additionally, it offers a quick demonstration of launching a micro-services app in just a few minutes using Docker Compose. The learning promises associated with this course encompass a wide range of topics. Participants will delve into micro-services and 12-factor app design, gaining the skills needed to run a micro-services app using Compose. The course also explores container orchestration, shedding light on its purpose and functionality. Container security best practices are covered to enhance participants' knowledge of making containers more secure. An overview of container logging and monitoring tools commonly used for tracking container performance is provided. The course concludes with an exploration of different container systems and operating systems, including Linux Containers such as LXC, Docker, rkt, OCI, and an overview of Windows Containers [25].

"DevOps on AWS Specialization" offered on Coursera, an intermediate-level program designed to equip learners with essential skills at the intersection of development and operations within the Amazon Web Services (AWS) environment. With a robust enrolment of 33,855 students and a stellar rating of 4.7 based on 1194 reviews.

This intermediate-level course "DevOps on AWS Specialization", distinct from the preceding container-centric courses, offers a more general focus on DevOps principles and practices within the AWS cloud environment. A notable factor is the absence of a prerequisite requirement for prior AWS cloud experience, as the first course within the specialization provides foundational knowledge. The specialization aims to equip participants with essential DevOps skills, encompassing continuous integration, DevOps practices, monitoring, logging, continuous delivery, and the deployment of micro-services. As part of the comparative analysis, it is essential to draw distinctions between the specific learning objectives and outcomes of this course and the container-focused courses analysed in this research. Emphasizing the variances in content, prerequisites, and focus, particularly in terms of cloud deployment and DevOps practices, will facilitate a comprehensive comparison between these two types of courses. The container courses' primary focus is on containerization technology, while this specialization takes a broader approach, concentrating on DevOps principles within the AWS environment [30].

Introduction to DevOps available on Coursera, a beginner-level course aimed at providing foundational knowledge and practical skills in the field of DevOps. With an impressive enrolment of 109,278 students and an outstanding rating of 4.8 based on 2440 reviews, this course is highly regarded by learners seeking to kick-start their journey into the world of DevOps. Through this course, participants gain a comprehensive understanding of DevOps principles, practices, and tools, empowering them to effectively collaborate between development and operations teams to deliver high-quality software products.

"Introduction to DevOps" beginner-level course serves as an entry point for individuals new to the world of DevOps, making it accessible for those with basic IT literacy, while also catering to learners seeking to expand their existing knowledge of DevOps practices. The primary objectives of the course are to impart fundamental skills in the domains of cloud-native computing, DevOps principles, continuous integration and continuous delivery (CI/CD), agile methodologies, and test-driven development (TDD) and behaviour-driven development (BDD). It is worth noting that while this course has a broader DevOps focus, it might encompass container technology as part of its teachings. Containerization technology, particularly in the context of micro-services, automated deployments, and building resilient code, often aligns with DevOps practices. The course promises to convey essential characteristics of DevOps, emphasizing the development of a culture characterized by shared responsibility, transparency, and embracing failure. Learners can expect to delve into pivotal DevOps concepts, including software engineering practices, cloud-native micro-services, automated continuous deployments, and the creation of resilient code. The course also addresses the significance of CI/CD, Infrastructure as Code (IaC), and TDD/BDD, offering insights into these critical DevOps practices. Additionally, it delves into the organizational impacts of DevOps, which includes breaking down silos, fostering collaboration in cross-functional teams, and shared responsibilities [31].

"DevOps, Cloud, and Agile Foundations Specialization" available on Coursera, designed for beginner-level learners aspiring to grasp the fundamental concepts of DevOps, cloud computing, and Agile methodologies. With a significant enrolment of 13,486 students and an impressive rating of 4.8 based on 931 reviews, this specialization is highly acclaimed for its comprehensive coverage and practical approach to DevOps, cloud technologies, and Agile practices.

"DevOps, Cloud, and Agile Foundations Specialization" beginner-level specialization is tailored to accommodate a diverse range of learners, requiring only basic computer literacy. It is designed to be accessible to individuals with or without a college degree and those without prior programming experience. The primary objectives of this specialization are to equip participants with fundamental skills in the domains of cloud-native computing, DevOps practices, Scrum methodologies, cloud computing, and agile principles. While this specialization has a broader focus, it could potentially include components related to container technology as part of its teachings, particularly in the context of DevOps practices. Containerization, when integrated with cloud-native computing and DevOps, becomes a key technology. The specialization promises to offer a comprehensive understanding of DevOps, encompassing essential characteristics, including culture, behaviour, practices, tools, methodologies, technologies, and metrics. Participants can anticipate an exploration of core concepts of cloud computing, including its benefits, models, use cases, platforms, services, resources, and opportunities. Furthermore, agile practices, such as small batches, minimum viable products, pair programming, behaviour- and test-driven development, are central to the curriculum. The specialization also delves into Scrum methodologies, equipping learners with the skills to create a product backlog, write user stories, utilize the sprint planning process, and measure performance [32].

"IBM DevOps and Software Engineering Professional Certificate" offered on Coursera, tailored for beginners seeking to venture into the realms of DevOps and software engineering. With a substantial enrolment of 14,493 students and a commendable rating of

4.7 based on 3,014 reviews, this professional certificate program is esteemed for its comprehensive curriculum and hands-on approach to DevOps and software engineering practices.

Designed for beginners "IBM DevOps and Software Engineering Professional Certificate" and those looking to expand their knowledge of DevOps, this professional certificate program encompasses a comprehensive curriculum and assumes basic IT literacy as a prerequisite. The primary objectives of this program are to empower participants to master the domains of DevOps, Agile methodologies, Scrum practices, continuous integration and continuous delivery (CI/CD), and cloud-native software engineering, ensuring they acquire hands-on, job-ready skills. A noteworthy aspect of this program is its direct alignment with container technology and cloud-native practices, furthering the understanding of DevOps within the context of modern software engineering. The program encourages the development of a DevOps mind set while fostering the practice of agile philosophy and Scrum methodologies, skills considered essential in the era of cloud-native software engineering. Participants will engage in the creation of applications using Python, leveraging various programming constructs and logic, such as functions, REST APIs, and libraries. The program stands out for its focus on constructing applications using micro services and deploying them with container technologies such as Docker, Kubernetes, and OpenShift, along with delving into server less technologies. Moreover, learners will employ a range of tools for automation, continuous integration (CI), and continuous deployment (CD), including popular options like Chef, Puppet, GitHub Actions, Tekton, and Travis [33].

These courses demonstrate the practical incorporation of container technology into DevOps education, offering participants a robust foundation in containerization and its role in modern software engineering. The inclusion of container-related content enhances the program's job-readiness, providing learners with valuable skills in demand in the software industry. This exemplifies how professional certificate programs like this bridge the gap between educational content and industry requirements.

3.2 Job analysis

In this research, an analysis of the professional job market has been conducted, with a particular focus on understanding the demand for container-related skills within the industry. The primary objective of this analysis is to discern the extent to which container technology is sought after and at which proficiency levels it is required across various job roles and sectors. This approach ensured that the analysis captured relevant job postings across various industries and sectors, particularly those in software development, DevOps, and cloud computing. By targeting roles where containerization skills are highly valued, the analysis aimed to provide a comprehensive overview of the demand for container expertise in the contemporary job market. The manual curation of job advertisements allowed for a detailed examination of the prominence of container-related skills, shedding light on their essentiality and the proficiency levels sought by employers

By examining a wide range of job advertisements from a perspective, valuable insights will be provided into the prominence of containers in the modern workforce, offering an understanding of their essentiality, the industries that prioritize them, and the associated skill levels. This research aims to offer comprehensive insight into the significance of container expertise in the job market and guidance for educational programs and newly graduated professionals looking to align their skill sets with industry demands.

3.2.1 Analysis of Job advertisements

In the job market analysis performed on LinkedIn, the job role for the title of Senior Software Developer with a focus on data and analytics, the requirement for container competence is explicitly expressed through the mention of "containers e.g. Kubernetes." This highlights the importance of container knowledge, with Kubernetes serving as a significant container orchestration platform. The job description also emphasizes the need for a comprehensive skill set alongside container expertise. Specific software development proficiency, both in backend and front-end development, is required. Additionally, hands-on experience with technologies like Java, Spring Boot, Azure, and DevOps reflects the importance of full-stack development and automation in the role. The focus on developing data-driven solutions aligns with the growing trend of deploying data-related applications in containerized environments. This confluence of container expertise with various software development competences underscores the value of a well-rounded skill set in the job market and demonstrates the relevance of container-related courses in preparing graduates for these in-demand positions [34].

In the job market analysis performed on LinkedIn, the job role for the title of “DevOps Engineers” serves as a valuable case study in how required container competence is expressed, alongside other competencies typically sought in combination with container expertise. Notably, the requirement to "work with containers in a distributed micro-services environment" distinctly highlights the need for container proficiency, emphasizing their pivotal role in enabling micro-services deployment and scalability. Mentioning of "Cloud Azure" reflects an indirect connection to container technology, as Azure often employs Kubernetes for container orchestration. This suggests the importance of managing containers in a cloud environment like Azure, underlining the intersection of container expertise with cloud management. In addition to container skills, the job posting encompasses a wide array of competencies. These include platform development, application migration to the cloud, deployment automation, Linux operating system proficiency, relational databases, Azure account provisioning, infrastructure monitoring, and familiarity with modern DevOps tools. While not all of these directly pertain to containers, their inclusion highlights the integrative nature of container technology in various aspects of contemporary software development, infrastructure management, and DevOps practices. This analysis accentuates the demand for container expertise in concert with a broad skill set, underlining the pertinence of container-related courses in preparing graduates for multifaceted roles in the job market [35].

A job posting under analysis from LinkedIn illustrates the demand for a Data Engineer or DevOps Engineer with a multifaceted skill set. Notably, the requirement for experience in DevOps, infrastructure as code, and containers demonstrates a strong emphasis on container technology, underlining its significance in modern software development. Additionally, proficiency in SQL and Python, with the potential inclusion of Scala and data-intensive programming tools, is regarded as valuable. The role necessitates experience in crafting production-grade ETL pipelines, particularly on distributed systems, and proficiency in data pipeline orchestration systems such as Apache Airflow. Familiarity with cloud services from major providers, including AWS, GCP, or Azure, is essential. Furthermore, the role delves into the domain of MLOps, emphasizing data and ML model versioning. Effective communication and the ability to work both autonomously and as part of a team are essential attributes for this position. In this analysis, the emphasis on container expertise, along with a broad array of skills spanning data engineering, cloud services, and DevOps practices, underscores the integrated role of containerization within the contemporary job market. This analysis provides valuable insights into the interplay between container expertise and a comprehensive skill set in roles like Data Engineering and DevOps [36].

The vacancy description for a Cloud Support Engineer outlines a thorough set of abilities and knowledge required for this position. It primarily focuses on hands-on experience in supporting Linux systems, emphasizing proficiency in virtualization and cloud technologies such as KVM and OpenStack, containerization with Docker, LXD/LXC, or Kubernetes, and a deep understanding of storage technologies encompassing block, object, and network storage. Networking skills, including bonding, firewalling, bridging, switching, and network file system tuning, are also critical. Proficiency in Linux integration with other environments, cloud computing, and troubleshooting, ranging from reading stack traces to understanding OS and application-level bugs, is crucial. Additionally, a foundational knowledge of programming in any language is valued, coupled with strong customer support skills, making customer needs a top priority, and clear and effective communication in English. This job posting underlines the importance of a well-rounded skill set in a Cloud Support Engineer role, encompassing container expertise alongside a range of other technical and communication skills, offering valuable insights into the intersection of container technology with cloud support roles in the job market [37].

Another role description of a DevOps Engineer encompasses a wide array of skills and competencies vital for optimizing development and operational processes on LinkedIn. Key requirements include experience with Infrastructure as a Service (IAAS) providers such as AWS, GCP, or Azure, and proficiency in Terraform, Ansible, Chef, or Puppet. Moreover, the position entails expertise in virtual networking, covering VPCs, subnets, firewalling, routing, and NAT, as well as a working knowledge of load balancers and domain services. Familiarity with security best practices, identity and access tools, secrets management, and Continuous Integration and Delivery (CI/CD) tools is paramount. Proficiency in source control, software configuration, and strong scripting skills in Bash, Ansible, Python, or Golang is expected. A strong customer-centric focus, collaboration within a team, and effective communication skills are also essential. While expertise in containers and Kubernetes is a "nice-to-have," the role emphasizes a passion for open-source software and a continuous desire to enhance technical and interpersonal skills. This job advertisement underscores the multifaceted nature of DevOps roles, with container and open-source proficiency serving as valuable assets in the contemporary job market [38].

3.3 Student Survey

A thorough student survey was carried out in an attempt to get important insights into the viewpoints and experiences of students about the teaching of container technology. The main objective of the survey was to gather opinions from people who had completed courses on containers or were already familiar with them. The surveys were administered through an online platform [39], specifically Google Forms. The survey consisted of a total of eight questions. The questionnaire encompassed the following inquiries.

- Level of education and field of study of the participant.
- Familiarity with container technologies (e.g., Docker, Kubernetes)?

Also, In addition to past encounters with the query, do you have experience using Containers prior to beginning of this course, multiple-choice questions were also asked for additional analysis.

- Received formal education/Training on container technologies as a part of your educational program/Self-study from online courses
- Educational program prepared you for working on container technologies in professional setting
- Opportunity to work on real world projects

Towards the conclusion of each survey, an open-ended question was included [38].

- Challenges encountered in learning container technologies

About 22 responses were gathered from the nearly 60 people who were contacted. A wide range of perspectives were provided by these answers, the majority of which were from the Information and Communication Technology (ICT) sector. The remaining respondents were people who were currently working with containers or had varied levels of experience with them. This survey is an attempt to better understand the container technologies education environment and collect the opinions of people who have already experienced this dynamic.

Participants were given eight carefully considered questions in total [40]. The questionnaire encompassed a wide range of topics, such as the participants' educational background and degree level in addition to six quantitative questions designed to assess their knowledge, opinions, and experiences with container technology. One especially perceptive question invited participants to discuss their own experiences and the difficulties they had when learning about containers, delving into the qualitative domain. Respondents were prompted by this open-ended question to be frank and insightful in their answers, which helped us shed light on the particular challenges that different students and persons encounter when learning about container technology. This survey's mix of quantitative and qualitative questions was intended to give respondents a full understanding of the topic and a dataset that will greatly contribute to the ongoing research on the educational challenges of container technology.

3.4 Survey of Cloud technology Education Research Reports

This comprehensive literature review embarks on a thorough exploration of the transformative impact of cloud technologies and containerization on education. Through the evaluation of five distinct research reports, pivotal questions regarding the integration, practical implications, and responses to these technologies are addressed. The assessment begins with an examination of the Career and Technical Education (CTE) 2018 workshop, shedding light on the discussions surrounding the integration of cloud technologies into education and uncovering valuable insights from this global forum. Subsequently, the review delves into the role of cloud technologies in STEM education, emphasizing their potential in robotics exploration and the organizational factors influencing their successful integration. A focused evaluation of a remote learning course implemented during quarantine provides practical insights into the challenges and adaptations in the educational landscape. Shifting the focus to higher education, the review explores the transformative impact of cloud technologies in creating flexible learning environments and highlights the frameworks/models developed to manage their implementation. Lastly, the examination of emerging roles in cloud technology, specifically containers and Docker, unveils their transformative potential and the responses of industry leaders. Together, these evaluations offer a nuanced understanding of the multifaceted implications of cloud technologies and containerization in shaping modern education

3.4.1 How cloud technologies continues to transform education

The CTE 2018 workshop emerged as a pivotal gathering that shed light on the ever-evolving landscape of cloud technologies within the realm of education. This distinct event provided a vibrant platform where a diverse range of thought leaders, including researchers, practitioners, and educators, convened to participate in profound discussions regarding the latest innovations and the complex array of challenges that define the dynamic sphere of educational technology. Drawing attendees from around the world, the workshop boasted a diverse range of participation, with contributors and presenters coming from various nations, including Israel, Poland, Sweden, and Ukraine. These individuals came together to exchange their invaluable insights, contributions, and discoveries within the extensive field of cloud technologies in education, fostering a truly global exchange of knowledge and experiences. At its essence, the workshop had a resolute goal to advocate for the integration of cloud technologies into educational practices and unlock the untapped potential they possess. With a strong focus on enhancing teaching and learning experiences, the workshop aimed to explore the transformative capabilities that cloud technologies offer. The result was a diverse exploration of a wide range of topics, with the papers presented at the workshop touching on various areas of educational research. These included the use of cloud technologies to enhance media literacy in preschool education, the innovative approaches to training translators in the age of information, the critical organization of scientific information, and the integration of e-learning courses as powerful tools for teaching mathematical disciplines. One of the standout findings from the workshop was the remarkable effectiveness of web 2.0 technology tools, such as educational blogs, in the field of foreign language teaching systems. The implementation of these tools resulted in significant benefits, including notable improvements in teaching effectiveness, the optimization of student performance, and the development of essential communicative and technological skills. In summary, the CTE 2018 workshop served as a dynamic melting pot where researchers and educators came together to explore the limitless potential of cloud technologies in the field of education. The knowledge shared and experiences exchanged covered a variety of domains, including the crucial areas of media literacy, language instruction, and scientific information management, among others. As a result, this workshop provided a profound tapestry of insights and learning, marking a pivotal moment in the ongoing evolution of educational technology [41].

3.4.2 Cloud technologies for STEM eduction

Cloud technologies have garnered a significant position within the domain of STEM education, providing students with an avenue to explore the captivating realm of robotics. This adoption not only holds the potential to augment system interfaces but also to render learning more accessible. However, the effectiveness of training future educators through cloud technologies is a matter of utmost importance, serving as the impetus that mold the content and essence of educational activities. The successful integration of cloud technologies into education depends on several organizational factors. These encompass the proficiency of educators in information and communication technologies (ICT), the existence of the requisite technological infrastructure, and the critical realignment of curricula to accommodate the advantages of cloud-based learning. Indeed, these factors play a pivotal role in ensuring the seamless incorporation of cloud technologies into the fabric of education. While cloud technologies offer a plethora of benefits, it is crucial to acknowledge the limitations they entail. These include a certain degree of reliance on the availability of electricity, as well as concerns regarding data control, often held by the cloud service provider. Nevertheless, these challenges do not diminish the significance of cloud technologies within the educational landscape. In fact, cloud technologies serve as a linchpin in the realization of contemporary educational trends. They empower the pursuit of interdisciplinary studies, foster collaboration, unlock a wealth of educational resources, and pave the way for innovative learning approaches. Consequently, cloud technologies have become an integral cornerstone of the evolving educational landscape, opening up new possibilities and reshaping our approach to learning and teaching [42].

3.4.3 Features of distance learning of cloud technologies for the organization educational process in quarantine

The course entitled "Cloud Technologies in Remote Learning during Quarantine" was deliberately developed to acquaint educators with the fundamental capabilities of cloud technologies in the context of remote education. This all-encompassing course covers a broad range of subjects, spanning from the basic principles and models of cloud services to the architecture and offerings from leading cloud service providers. Participants delve into practical aspects, such as the management of electronic mail, storage based on cloud technology, collaborative document creation, online surveys, interactive presentations, and creation of intelligent maps, website development, and the utilization of cloud-based learning management systems such as Google Classroom. Throughout the duration of the course, educators acquire first-hand experience in creating presentations utilizing cloud technologies, while also being exposed to a carefully selected assortment of cloud services specifically tailored to the requirements of the course. To facilitate effective learning, the course is delivered in three distinct phases, each of which spans a duration of five days. This approach ensures that participants quickly acquire the essential skills necessary to harness the potential of cloud technologies for educational purposes. Nevertheless, the journey of mastering cloud technologies is not without its difficulties. Participants encounter limitations pertaining to time, the transition from traditional teaching methods to remote learning, and the intricate task of reorganizing the educational process in response to quarantine measures. These obstacles are an integral part of the learning process and provide valuable insights into the dynamic landscape of cloud-based education [43].

The decision to carry out a survey on the Quarantine paper in education originated from a deep realization that the world, post-quarantine, is swiftly moving towards the domain of online education. The global panorama has witnessed an unparalleled acceleration in the acceptance of online learning platforms and digital educational tools as a response to the challenges imposed by quarantine measures. With this transformative shift, the significance of comprehending the dynamics, difficulties, and advancements in remote education has become of utmost importance. The survey aimed to capture the essence of this pivotal transition and illuminate the evolving educational paradigms, providing insights that are of great value for educators, institutions, and learners as they navigate the unexplored territory of a digitally-driven educational landscape.

3.4.4 Cloud Computing Adoption in Higher Education Institutions, A Systematic Review

Cloud technologies have emerged as a dynamic force within higher education institutions (HEIs), facilitating the creation of flexible and mobile learning environments. These technologies empower educators to embark on an innovative journey, integrating various cloud-based tools such as Google Apps, Facebook, and mind-mapping applications to construct novel and immersive teaching and learning spaces. Consequently, this has led to the establishment of highly customizable personal learning environments, where virtual desktops equipped with pre-installed software play a pivotal role in shaping the educational landscape, particularly for students enrolled in computing and technology courses. In addition to transcending the boundaries of traditional learning, cloud technologies extend their capabilities to cater to the unique learning requirements of students with ADHD, offering augmented reality experiences. Furthermore, these technological advancements enhance the mobile learning experience by providing increased processing power and data storage capacity. To effectively manage the intricacies of these transformative technologies, meticulous frameworks and models have been developed, enabling HEIs to administer their systems and cultivate cloud-based learning environments. It is noteworthy that while extensive research attention has been dedicated to the realm of e-learning, cloud-based mobile learning and collaborative learning systems have received relatively less scholarly investigation. This presents an exciting frontier for future exploration, as these technologies continue to reshape the landscape of higher education and the broader domain of learning [44].

3.4.5 Containers & Docker Emerging Roles & Future of Cloud Technology

Technologies such as Docker serve as examples of container-based virtualization, which is a rapidly emerging force that will have a significant impact on the future of cloud computing. This paradigm shift, which is quickly changing the digital world, predicts a future in which containers are positioned to form the foundation of micro service applications. Reputable cloud providers are at a crossroads where they must decide whether to adopt a new container stack or add support for containers to their current architecture. Cloud service providers are also experiencing a transformational wave; major players in the area, including IBM, Amazon Web Services (AWS), Joyent, and IBM, Google, Rack space, are all expanding their Container-as-a-Service (CaaS) offerings.

The prevailing interest in Docker and container technology among established participants in the technology ecosystem has also prompted legacy businesses to launch container-based products in response to the growing demand for and acceptance of these new paradigms. Prominent companies like VMware, IBM, and Microsoft are actively developing their container strategies in the vast cloud space. In the midst of all of this upheaval, the Open Container Project (OCP) stands out as an important endeavour since it aims to guide the IT industry towards better container deployment practises. Leading industry players including as VMware, Amazon Web Services, IBM, Microsoft, Google, EMC, and Red Hat are involved in this joint project, demonstrating the industry's shared dedication to the advancement of container technology. In summary, the data presented here highlights three main themes, the widespread adoption of container-based virtualization, the careful development of container strategies by industry leaders, and the critical role that containers will play in cloud technology going forward. This path of transformation is leaving a trail of significant changes that will likely redefine the future of digital infrastructure [5].

This integrated evaluation of the research reports reveals a cohesive narrative on the transformative influence of cloud technologies and containerization in education. The CTE 2018 workshop serves as a global hub where educators discussed innovative applications, showcasing the effectiveness of web 2.0 tools in foreign language teaching systems. The report on STEM education emphasizes the accessibility and potential of cloud technologies in robotics exploration, underlining the organizational factors crucial for their successful integration. The examination of a remote learning course during quarantine demonstrates the practical implementation of cloud technologies in education, offering insights into the challenges and adaptations in the evolving educational landscape. Reports on higher education institutions highlight the flexible and personalized learning environments facilitated by cloud technologies, particularly through virtual desktops. Lastly, the exploration of emerging roles in cloud technology, such as containers and Docker, unveils their transformative potential and industry responses. Common themes include a focus on transformative impacts, challenges in integration, and the evolving educational landscape, contributing to a nuanced understanding of the multifaceted role of cloud technologies and containerization in shaping modern education. In conclusion, these reports review provides a holistic understanding of the multifaceted impact of cloud technologies on education, uncovering transformative trends, challenges, and avenues for future exploration in the realm of containerization and cloud technology.

4. RESULTS

Based on the analysis performed with respect to educational programs, job market as well as the self-conducted student survey, the following findings were found which can help provide an insight into the challenges faced by educational sector regarding containerization. By delving deeper into these insights, we can gain a clearer understanding of the complex issues confronting educational institutions and their efforts to adapt to the evolving demands of the containerization industry. These findings serve as a valuable resource for identifying potential solutions and opportunities for improvement within the educational sector in response to the challenges presented by containerization.

4.1 Challenges with respect to educational programs and Job market

In the following section, we present the results of our extensive research and analysis, which have illuminated key insights into the dynamic landscape of containerization education and its alignment with the ever-evolving job market demands. By carefully examining the data obtained from our evaluation of container-related courses, the comparison between traditional and container-focused programs offerings, and the scrutiny of job market requirements, we have unveiled a series of critical findings. These findings serve as a bridge between the realms of education and employment, shedding light on the challenges and gaps that exist within the educational sector concerning containerization. Our results provide a comprehensive overview of the current state of container education, the skills needed in the job market, and the pivotal areas where improvements and solutions are warranted. Below are a challenges that has been found through the intensive study of containers educational program, traditional courses and job analysis.

No#	Challenges	Related academic Program	Related Online Course	Job market
1.	Comprehensive Education	Cloud Technologies and Systems Course	Introduction to Containers	Requires multifaceted knowledge of containerization, especially with Docker, Kubernetes, or other tools.

2.	Hands-on Skills Development	DevOps with Docker Course	Introduction to Containers	Data Engineer/DevOps Engineer: Emphasizes experience in container technology alongside other skills such as SQL, Python, and cloud services.
3.	Multi-platform Proficiency	Cloud Platforms Course	Basics of Using Containers in Production	Cloud Support Engineer: Requires proficiency in containers with Docker, LXD/LXC, or Kubernetes, along with other cloud technologies.
4.	Implementing cloud solutions via service configuration, Infrastructure as Code (IaC) methodologies, and script development poses a significant challenge.	Cloud Technologies and Systems	DevOps with Docker	The job market analysis revealed requirements for roles like DevOps Engineers, which necessitate experience in DevOps, infrastructure as code, and containers. This emphasizes the importance of container technology alongside a broader array of skills in modern software development roles.
5.	Lack of practical project components in container educational programs leads to insufficient	Continuous Development and Deployment - DevOps	Introduction to Containers with	The job market analysis indicates a demand for roles like DevOps Engineers,

	hands-on experience for learners.		Docker, Kubernetes & Open Shift	which require practical experience in containerization technologies. This underscores the importance of hands-on experience gained through educational programs in preparing students for roles in the job market.

Table 4. 1 Challenges with respect to educational programs and Job market

The primary challenge we uncovered is the absence of a comprehensive educational program for containerization. Current programs tend to focus on specific aspects, leaving gaps in students' understanding. This challenge is evident in the varying levels of confidence and preparedness among students, as indicated by our survey results. Students may not receive a well-rounded education in containerization, impacting their overall grasp of the subject.

A complete containerization program is important because it helps students understand how to use container technologies effectively. These technologies, like Docker and Kubernetes, are changing how we build and manage software. By studying containerization comprehensively, students learn not only the basics but also how to apply these skills in real-world situations. Since containerization connects with other areas like cloud computing and DevOps, a complete program ensures students have the diverse skills needed for today's tech jobs. In short, a complete containerization program prepares students to succeed in the fast-changing world of IT.

The need for hands-on skills development is met by academic programs like the "DevOps with Docker" course, which focuses on practical skills in containerization. Online courses such as "Introduction to Containers" also aim to equip learners with practical containerization skills. Job market analysis underscores the importance of container technology experience for roles like Data Engineers/DevOps Engineers. The challenge of multi-platform proficiency is tackled by academic programs like the "Cloud Platforms" course, which covers various cloud platforms. Similarly, online courses such as "Basics of Using Containers in Production" aim to provide knowledge across different container systems. Job market analysis highlights the importance of proficiency in container technology and other cloud technologies for roles like Cloud Support Engineers.

The fourth challenge focuses on the necessity to configure cloud solutions and establish Infrastructure as Code (IaC) methodologies. Educational programs frequently offer limited knowledge, potentially leaving students ill-equipped to handle the complexities of cloud solutions. Insights from our survey underscore the requirement for adaptable learning paces and hands-on experiences to bridge the gap. This challenge underscores the importance of incorporating comprehensive content on configuring cloud solutions and implementing IaC methodologies within educational programs.

Last but not least this challenge revolves around the lack of practical project components in container educational programs. This underscores the crucial role of hands-on experience in mastering container technology. Our survey unveiled a range of confidence levels among students, with some feeling adequately prepared for professional work, while others remained uncertain. This challenge highlights the necessity of incorporating practical, project-based components into container educational programs to ensure students gain the hands-on skills essential for success in the field.

4.2 Learning Challenges in Container Technology Survey

To gain a deeper understanding of the current learning obstacles in the ever-evolving field of container technology, a thorough survey of students was carried out [45].

1. Please indicate your current educational level (e.g., undergraduate, graduate) and your major or field of study?

Participants in the survey came from a variety of academic backgrounds. 12 were from programmes in software, web, and cloud; 2 were from degrees in computer science; 2 were from programmes in data science; 2 were from information technology (information security); and 1 was from Advanced Studies in Embedded Systems, HTI, Wireless Communication, and Automation Engineering. This wide range of responders offers our study a deep and complex viewpoint, illuminating the various obstacles students in a wide range of academic fields encounter when utilising container technology.

After the demographic data, the survey results are displayed in a series of graphs for different questions, giving participants a visual representation of the results from respondents.

2. How familiar are you with container technologies (e.g. Docker, Kubernetes)?

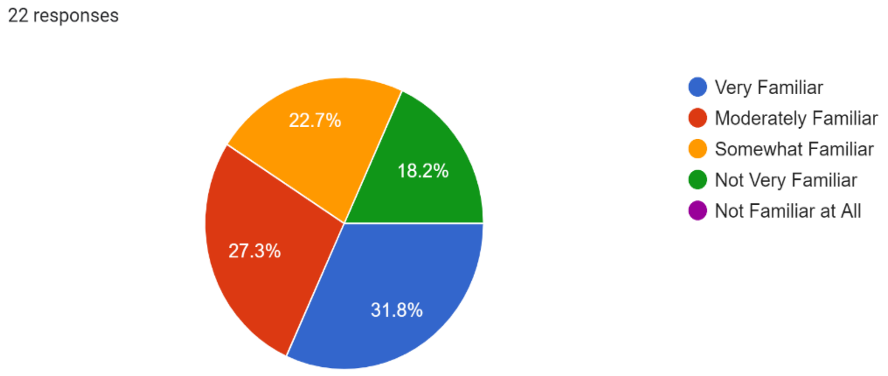

Figure 4.1 Respondents division based on familiarity of containers

A majority of the respondents expressed a reasonable level of familiarity with container technologies. Only 4 out of 22 respondents were unaware of the said container technology. 6 out of 22 were moderately familiar and 5 out of 22 were somewhat familiar.

3. Have you received formal education or trainging on containers technologies as part of your eductional program. Or did you learn about containers from other sources (e.g. online courses, self-study)?

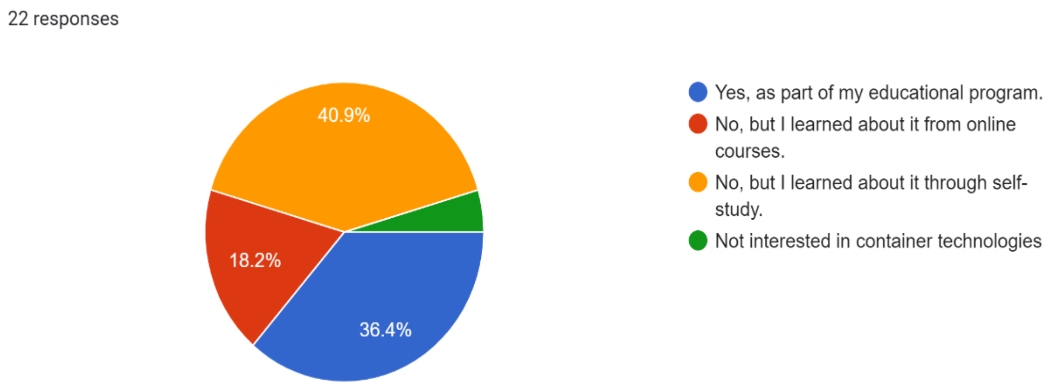

Figure 4.2 Respondents division based on containers education

The chart indicates that a significant portion, 36.3%, have received formal education or training as part of their educational program. Furthermore, 18.2% have independently sought online courses to enhance their understanding of container technologies, while 40.9% have chosen self-study as a means of learning. A notable 4.5% have expressed disinterest in container technologies. These responses provide valuable insights into the diverse learning pathways embraced by individuals, ranging from structured educational programs to self-initiated and online resources.

4. Rate your confidence in using containers technologies at your workplace from 1 (not confident) to 5 (confident)

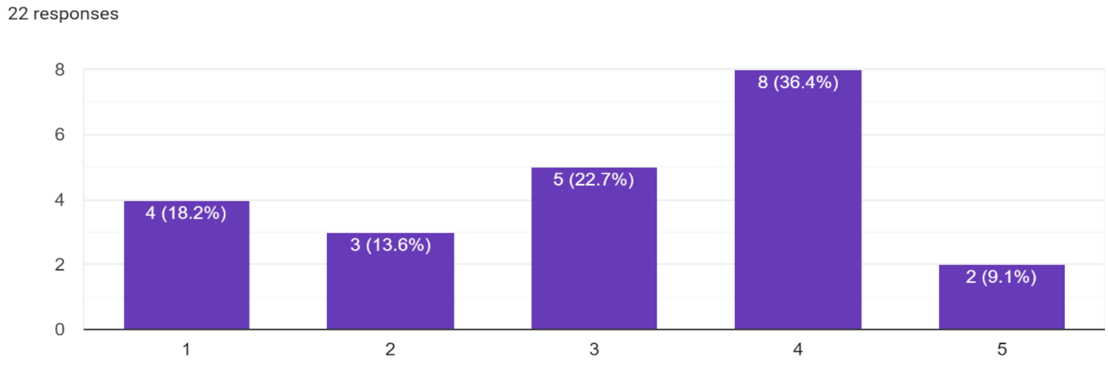

Figure 4.3 Respondents division based on using container at workplace

18.2% of respondents rated their confidence at level 1, indicating a lack of confidence.13.6% expressed a slightly higher level of confidence, rating themselves at level 2. A significant 22.7% reported a moderate level of confidence, rating themselves at level 3. The majority of participants, at 36.4%, placed themselves at level 4, signifying a high level of confidence. 9.1% of respondents, representing the smallest portion, rated their confidence at the highest level, 5, indicating a very high level of confidence in using container technologies in their workplace.

5. Do you believe your educational program adequately prepared you to work with container technologies in a professional setting?

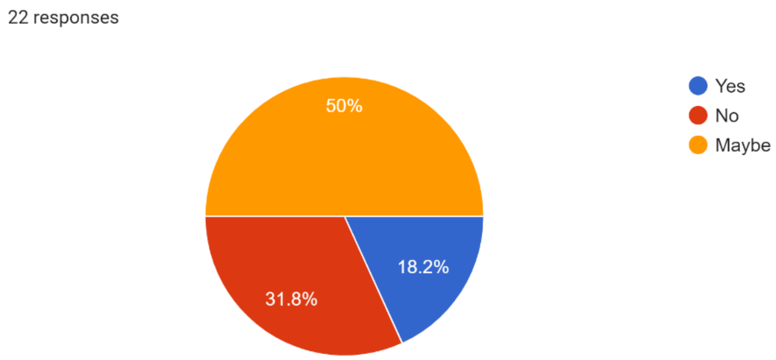

Figure 4.4 Educational Program & Professional Readiness

18.2% of respondents expressed a firm belief that their educational program had adequately prepared them for working with container technologies. On the other hand, 31.8% of participants indicated a negative response, believing that their educational program had not provided sufficient preparation. A significant 50% of respondents fell into the "Maybe" category, suggesting a degree of uncertainty about the adequacy of their educational program in relation to professional work with container technologies.

6. Do you feel that the pace of learning container technologies in your eductional programe is?

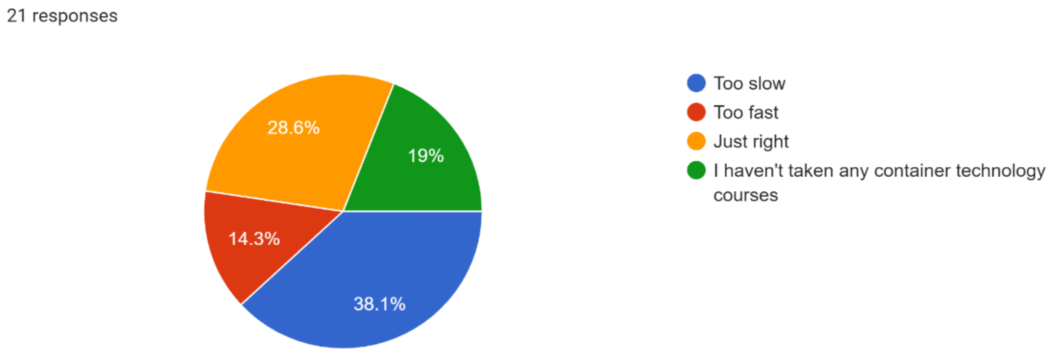

Figure 4.5 Respondents division based on pace of learning containers in educational programs

A considerable 38.1% of respondents find the pace of learning to be "Too slow," indicating a desire for a more accelerated learning process. In contrast, 14.3% believe the pace is "Too fast," suggesting a preference for a more measured approach. A substantial 28.6% of participants feel the pace is "Just right," indicating contentment with the current learning speed. A minority of 19% mentioned they haven't taken any container technology courses, thus exempting themselves from evaluating the pace of learning.

7. Have had the opportunity to work on real-world projects or apply container technology skills in practical scenarios as part of your educational program?

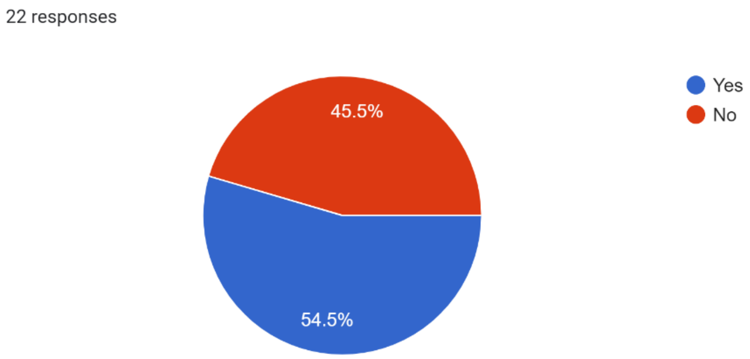

Figure 4.6 Respondents division based on opportunity to use containers in real world

A majority of 12 respondents (54.5%) have had the opportunity to work on real-world projects or apply container technology skills in practical scenarios within their educational program. Conversely, 10 respondents (45.5%) reported that they have not had such opportunities as part of their educational program.

Qualitative Analysis of Responses to the Open-Ended Question

8. What challenges have you encountered in learning about container technologies?

From the responses provided by 15 participants, several recurring themes and challenges in learning about container technologies have emerged.

Networking and Security Settings

Several respondents mentioned that understanding container networking and security settings presented challenges. This indicates the complexity of managing network configurations and ensuring security within containerized environments.

Lack of Practical Experience

Many participants highlighted the limited practical exposure they received, which hindered their ability to effectively use containers in real-world scenarios. Practical work, especially in large infrastructure deployments like Kubernetes, was deemed essential.

Evolving Ecosystem and Orchestration

Participants noted the evolving nature of the container ecosystem and the challenges associated with container orchestration, reflecting the dynamic nature of the field.

Resource Management

Balancing resource allocation, such as CPU and memory, within containers was identified as a specific challenge. Effective resource management is crucial for optimizing container performance.

Differences in Operating Systems

The variation in using container technologies across different operating systems, particularly Linux and Windows, was recognized as a challenge. This highlights the need for cross-platform skills.

Lack of Formal Education

Some respondents noted the absence of formal education or structured courses as a challenge in their container technology learning journey.

Limited Learning Resources

The limited availability of learning materials and support for container runtimes other than Docker was mentioned. This underscores the need for a broader range of resources.

Need for Technical Background

Several participants emphasized the importance of having a solid technical foundation and understanding real-world use cases to address the complexities of learning about containers.

Ease of Use on Different Platforms

Differences in user experience between Windows, Mac, and Linux systems were highlighted, indicating that the choice of the operating system could impact the learning experience.

Individual Trial and Error

Some participants acknowledged that they relied on trial and error to learn about containers due to a lack of a solid base. A basic understanding was seen as essential for a smoother learning journey.

The qualitative analysis reveals that learning container technologies is accompanied by multifaceted challenges. These challenges encompass technical aspects, such as networking and resource management, as well as the need for practical experience, a solid technical background, and the availability of comprehensive learning resources. Understanding the ever-evolving container ecosystem and addressing the differences in using containers across various operating systems are also noteworthy challenges.

To address these challenges effectively, educational programs and resources should consider providing more hands-on practical experiences, a stronger technical foundation, and accessible learning materials that cater to different container runtimes and platforms. Furthermore, acknowledging the dynamic nature of the container ecosystem and fostering a deeper understanding of real-world use cases can contribute to more successful container technology education (45).

5. DISCUSSION

Exploring the core of our research findings, interpretations, and insights in this crucial chapter, which sets the stage for a thorough examination of the consequences and importance of our work. The Discussion chapter helps us think deeply about the topic by analysing and combining ideas. It builds on what we've already discussed in earlier chapters. Here, the various threads of the research trip together, enabling us to make significant conclusions, useful suggestions, and an illumination of the work's wider significance. By having a meaningful conversation, the ability to address the research questions and provide insightful comments to both the academic and applied fields of our study. Together, the exploration of the nuances within the results and consideration of their broader implications will be undertaken as we embark on this intellectual journey. The ultimate goal is to advance the understanding and application within our field. In the dynamic landscape of containerization, there have been identification of several critical challenges that have far-reaching implications for both educational programs academic and non acamedic and the job market. Foremost among these challenges is the absence of a comprehensive and integrated educational program. Containerization is advancing rapidly, yet many educational programs are struggling to keep pace, providing only fragmented knowledge. While they may offer foundational insights, they often fail to address the multifaceted nature of containerization, encompassing micro-services, orchestration, and security. This leaves students ill-prepared for the ever-evolving job market, which demands expertise in all these areas.

The second challenge of hands-on skills development in container technology is multifaceted. Within academic programs, such as the "DevOps with Docker" course, emphasis is placed on practical experience in containerization, enabling students to gain valuable hands-on skills. These courses provide opportunities for students to implement and manage containers using Docker, fostering a deeper understanding of container technology. Similarly, online courses like "Introduction to Containers" complement academic learning by offering foundational knowledge and practical exercises in container technology. In the job market, roles like Data Engineer or DevOps Engineer highlight the significance of experience in container technology alongside other essential skills such as SQL, Python, and proficiency in cloud services. This underscores the importance of hands-on skills development in containerization to meet the demands of contemporary job roles effectively.

The third challenge of multi-platform proficiency presents a critical aspect in the realm of container technology. In academic programs, such as those focused on multi-platform proficiency, students are exposed to comprehensive training aimed at equipping them with expertise across various cloud platforms. Courses like "Basics of Using Containers in Production" provide foundational knowledge and practical insights into containerization, preparing students to navigate diverse cloud environments effectively. Complementing academic endeavours, online courses offer additional opportunities for learners to delve into container technology and its application in multi-platform scenarios. In the job market, roles such as Cloud Support Engineer underscore the significance of proficiency in container technologies like Docker, LXD/LXC, or Kubernetes, alongside a deep understanding of other cloud technologies. This highlights the demand for professionals who can seamlessly operate across multiple platforms, emphasizing the importance of addressing the challenge of multi-platform proficiency in both educational and professional settings.

The challenge we're looking at is all about the need to use cloud solutions effectively by setting up services, using Infrastructure as Code (IaC) methods, and creating scripts or automation tools. This is a significant challenge because it's essential in today's tech world. However, when it comes to educational programs, they often don't provide enough knowledge in this area. That means students may not get the skills they need for the job market, where expertise in cloud solutions is highly sought after. To tackle this challenge, we need to improve educational programs to offer more in-depth content, and students should consider further learning and training to meet the industry's demands. In a nutshell, this challenge is about making sure that what students learn matches with what employers are looking for in the field of cloud solutions.

The challenge we're addressing is the lack of hands-on experience in many container education programs, particularly the absence of practical project components. This is a significant issue because hands-on practice is crucial for learners in the field of containerization. Unfortunately, most courses don't include projects that offer this kind of practical experience. As a result, students may not have the hands-on skills needed for the job market, where in-depth experience is often a requirement. To tackle this challenge, it's essential to consider adding practical projects to educational programs to ensure students gain the experience they need. In summary, this challenge revolves around the need to provide practical, hands-on learning opportunities in container education programs to better prepare students for the job market's demands.

These challenges underscore the evolving nature of containerization education and its intricate relationship with the job market. Addressing these issues is crucial for better equipping students with the knowledge and skills they need to succeed in containerization-related roles.

To tackle these challenges, institutions/e-learning platforms must work towards developing integrated curricula that offer students a holistic understanding of containerization, which includes interdisciplinary modules, real-world case studies, and hands-on training. Such efforts ensure students graduate with well-rounded expertise. Furthermore, expanding cloud platform expertise offerings, integrating practical labs, and pursuing additional cloud certifications can bridge the gap between theoretical knowledge and practical cloud skills.

Enhancing practical experience in CI/CD processes and containerization requires institutions to integrate more hands-on learning opportunities, including real-world projects, simulations, and the use of industry-standard tools. Students can augment their education through internships, open-source contributions, and personal projects.

Addressing the absence of practical projects in container education programs requires a concerted effort. Institutions should consider incorporating hands-on projects that mimic real-world scenarios, providing students with the critical hands-on experience required in containerization roles.

Navigating the Learning Challenges in Container Technology Education

The "Learning Challenges in Container Technology Survey" engaged 22 participants from diverse academic backgrounds, including software, web, and cloud programs, computer science, data science, information technology (information security), and more. This diversity provided a comprehensive view of the challenges encountered by students in the pursuit of container technology education. The survey revealed a spectrum of confidence levels among participants in using container technologies in a workplace setting, with varying perceptions of their preparedness for professional work in the field. Notably, participants expressed diverse opinions on the pace of learning container technologies within their educational programs, highlighting the need for adaptable curricula.

Moreover, the survey underscored the division in participants' opportunities for practical exposure, with some having hands-on experiences and others lacking such opportunities. These findings emphasize the pivotal role of practical exposure in preparing students for professional roles and the importance of addressing gaps in experiences. Participants also voiced recurring themes and challenges in learning about container technologies, encompassing technical aspects, the need for practical experience, and resource management. The evolving container ecosystem, differences in using containers across operating systems, and limited learning resources were also identified as noteworthy challenges.

In light of these findings, there is a compelling case for the development of comprehensive educational programs, tailored curricula, and hands-on experiences. Providing a range of learning resources and fostering adaptable cross-platform skill sets is crucial to address the diverse challenges faced by students. This survey serves as a valuable foundation for future efforts to enhance container technology education, equipping individuals with the skills and knowledge needed to thrive in this dynamic field.

Within the broader landscape of container technology education, our exploration unveils a series of quintessential challenges and insightful findings. These collectively illuminate the multifaceted terrain of learning container technologies, providing valuable guidance for educators, program developers, and learners.

The challenges we have identified, ranging from holistic container education to the absence of practical project components, form the backdrop for this discussion. These challenges emphasize the vital importance of adaptive, comprehensive educational programs, practical experiences, and a diverse range of learning resources. They bridge the confidence gap and prepare students for professional roles in the dynamic container technology domain. Furthermore, our survey insights offer a comprehensive view of the hurdles encountered by students and learners pursuing container technology education. These findings underline the complex interplay between education and the ever-evolving field of container technology. The varying levels of confidence and preparedness displayed by students underscore the need for tailored curricula, adaptable learning paces, and practical exposure. As educators, program developers, and students continue to navigate this intricate educational landscape, the challenges and survey results discussed here will serve as crucial reference points. They offer the necessary direction for enhancing container technology education, enabling individuals to excel in this dynamic and ever-evolving field.

6. CONCLUSIONS

In this comprehensive exploration, our study has delved into the multifaceted landscape of containerization's role in both educational programs and the professional job market. This thesis embarked on a journey to dissect the educational content related to containerization, scrutinizing the practical skills students acquire. It traversed the realm of container creation and management, delving deeper into the orchestration of containerized applications. Moreover, our research ventured into how containerization augments students' grasp of pivotal software development methodologies, particularly Continuous Integration/Continuous Deployment (CI/CD) and DevOps practices. Simultaneously, we cast our gaze upon the job market landscape, unravelling the career opportunities that beckon those with containerization expertise, as well as the competencies sought by employers.

As the thesis unfolded, we discerned fundamental challenges that illuminate the intricacies of container technology education. The absence of holistic container education, the need for multifaceted understanding of cloud platforms, the complexities of navigating CI/CD for containerization, the imperative to configure cloud solutions and establish Infrastructure as Code (IaC) methodologies, and the absence of practical project components in educational programs each presented its unique dimension of complexity. These challenges reverberated in the voices of the students we surveyed, offering an enlightening perspective on the complex interplay between education and the evolving field of container technology.

Our findings underscore the significance of adaptive, comprehensive educational programs that cater to the diverse learning needs and paces of students. The landscape of containerization education, as illuminated by our research, calls for a multifaceted approach, one that bridges the confidence gap and prepares students for their professional roles in this dynamic domain. As we reflect upon the broader implications of this study, it becomes evident that the impact extends beyond education into the realms of the job market, software development, and IT infrastructure.

Educationally, our findings underscore the importance of creating educational programs that address the unique challenges of container technology education, ensuring that students graduate with not only theoretical knowledge but also practical skills. This aligns with the requirements of the job market, which consistently seeks professionals who can navigate the complexities of containerization and contribute effectively to their organizations.

In the job market, containerization expertise is an asset, and our study serves as a guide for both job seekers and employers. Students, equipped with practical knowledge, can confidently pursue careers in container technology, aware of the competencies sought by employers. Employers, on the other hand, gain insights into the capabilities of candidates, understanding that the challenges students face during their educational journey contribute to their unique skill set.

The broader industry, comprising software development and IT infrastructure, is not immune to the transformative impact of container technology. Our study offers a glimpse into the future of these domains, where containerization is poised to play an increasingly significant role. By understanding the challenges faced by students, educators, and employers, the industry can adapt, ensuring that the transition to container technology is seamless and productive.

As we conclude this study, we acknowledge its limitations. The landscape of container technology is constantly evolving, and our findings capture a specific moment in time. To maintain their relevance, educators and job seekers must stay updated on the latest developments in container technology and adapt accordingly. Future research should aim to provide a dynamic perspective, tracking the evolving educational and professional landscape.

In summary, our thesis has endeavoured to provide a holistic and multi-vocal understanding of containerization's impact on education and the job market. It serves as a comprehensive resource, benefiting educators, students, and professionals alike. By systematically implementing the research steps outlined in this study, we hope to enrich the discourse on containerization's role and potential within the domains of software development and IT infrastructure. It is our belief that this research serves as a cornerstone, facilitating the adaptation of education and the job market to the ever-evolving container technology landscape.

REFERENCES

1. Koskinen M, Mikkonen T, Abrahamsson P. Containers in Software Development: A Systematic Mapping Study. In: Franch X, Männistö T, Martínez-Fernández S, editors. Product-Focused Software Process Improvement [Internet]. Cham: Springer International Publishing; 2019 [cited 2023 Oct 10]. p. 176–91. (Lecture Notes in Computer Science; vol. 11915). Available from: http://link.springer.com/10.1007/978-3-030-35333-9_13

2. Cloud Computing [Internet]. [cited 2024 Mar 12]. PaaS and containers: Key differences, similarities and uses | TechTarget. Available from: https://www.techtarget.com/searchcloudcomputing/feature/PaaS-and-containers-Key-differences-similarities-and-uses

3. Engineering Education (EngEd) Program | Section [Internet]. [cited 2023 Oct 10]. A Brief History of Container Technology. Available from: https://www.section.io/engineering-education/history-of-container-technology/

4. IT Operations [Internet]. [cited 2023 Nov 14]. The evolution of containers: Docker, Kubernetes and the future | TechTarget. Available from: https://www.techtarget.com/searchitoperations/feature/Dive-into-the-decades-long-history-of-container-technology

5. Singh S, Singh N. Containers & Docker: Emerging roles & future of Cloud technology. In: 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) [Internet]. 2016 [cited 2023 Oct 10]. p. 804–7. Available from: https://ieeexplore.ieee.org/document/7912109

6. The Good and the Bad of Docker Containers [Internet]. AltexSoft. [cited 2023 Oct 4]. Available from: https://www.altexsoft.com/blog/docker-pros-and-cons/

7. Kubernetes [Internet]. [cited 2023 Oct 19]. Production-Grade Container Orchestration. Available from: https://kubernetes.io/

8. Casalicchio E. Container Orchestration: A Survey. In: Systems Modeling: Methodologies and Tools [Internet]. 2019. p. 221–35. Available from: https://www.researchgate.net/publication/328345589_Container_Orchestration_A_Survey

9. Mikkonen T, Pautasso C, Systa K, Taivalsaari A. Cargo-Cult Containerization: A Critical View of Containers in Modern Software Development. In: 2022 IEEE International Conference on Service-Oriented System Engineering (SOSE) [Internet]. Newark, CA, USA: IEEE; 2022 [cited 2023 Oct 10]. p. 93–8. Available from: https://ieeexplore.ieee.org/document/9912622/

10. Rosen C. Containers are Efficient—Now, Make Them Even Better [Internet]. IBM Blog. 2019 [cited 2023 Nov 14]. Available from: https://www.ibm.com/blog/containers-are-efficient-now-make-them-even-better/

11. Snyk [Internet]. 2022 [cited 2023 Nov 14]. Best practices for container isolation. Available from: https://snyk.io/blog/best-practices-for-container-isolation/

12. Middleware [Internet]. [cited 2023 Nov 14]. Understanding the Docker Ecosystem: A Basic to Advance Guide. Available from: https://middleware.io/blog/understanding-the-docker-ecosystem/

13. Container Ecosystem And Its Significance In Cloud [Internet]. [cited 2023 Oct 10]. Available from: https://coreitx.com/blogs/what-is-the-container-ecosystem-in-cloud

14. Aqua [Internet]. [cited 2023 Nov 7]. 3 Types of Container Runtime and the Kubernetes Connection. Available from: https://www.aquasec.com/cloud-native-academy/container-security/container-runtime/

15. What is a container registry? [Internet]. [cited 2023 Nov 7]. Available from: https://www.redhat.com/en/topics/cloud-native-apps/what-is-a-container-registry

16. What is container orchestration? | IBM [Internet]. [cited 2023 Nov 7]. Available from: https://www.ibm.com/topics/container-orchestration

17. Mehar A. AlmaBetter. 2023 [cited 2023 Nov 7]. Docker Architecture and Its Components: Detailed Explanation. Available from: https://www.almabetter.com/bytes/articles/docker-architecture

18. Log Analysis | Log Monitoring by Loggly [Internet]. [cited 2023 Nov 7]. Container Monitoring in Modern IT Environments - Guide | Loggly. Available from: https://www.loggly.com/use-cases/container-monitoring-in-modern-it-environments-guide/

19. Aqua [Internet]. [cited 2023 Nov 14]. What is Container Compliance? - Aqua Security. Available from: https://www.aquasec.com/cloud-native-academy/container-security/container-compliance/

20. Kinetics C. CI/CD Pipeline For Container-Based Workloads [Internet]. Cloud Kinetics. 2022 [cited 2023 Nov 14]. Available from: https://www.cloud-kinetics.com/blog/enabling-ci-cd-pipeline-for-container-based-workloads/

21. Cloud Academy [Internet]. [cited 2023 Oct 19]. Introduction to Containers Course. Available from: https://cloudacademy.com/course/introduction-to-containers/introduction-77/

22. Udemy [Internet]. [cited 2023 Oct 19]. Introduction to Containers - 90 Minute Crash Course. Available from: https://www.udemy.com/course/containercrashcourse/

23. Introduction to Containers w/ Docker, Kubernetes & OpenShift | Coursera [Internet]. [cited 2023 Oct 19]. Available from: https://www.coursera.org/learn/ibm-containers-docker-kubernetes-openshift?action=enroll#recommendations

24. The Twelve-Factor App [Internet]. [cited 2024 Feb 27]. Available from: https://12factor.net/

25. Cloud Academy [Internet]. [cited 2023 Oct 19]. Basics of Using Containers in Production Course. Available from: https://cloudacademy.com/course/basics-of-using-containers-in-production/introduction-83/

26. Cloud Technologies and Systems [Internet]. [cited 2024 Jan 25]. Available from: https://sis-tuni.funidata.fi/student/courseunit/otm-86711f21-1ea2-44f3-bd28-52c196e96400/brochure

27. Course Unit TKT21036 DevOps with Docker 2023–2026 | Courses | Studies Service | University of Helsinki [Internet]. [cited 2024 Feb 27]. Available from: https://studies.helsinki.fi/courses/course-unit/otm-66a41e7a-7bc5-4619-bccf-afa2474dcf04/TKT21036

28. COMP.SE.140: Continuous Development and Deployment - DevOps | Tampere University [Internet]. [cited 2024 Jan 25]. Available from: https://www.tuni.fi/en/students-guide/curriculum/course-units/otm-f70f0faa-127a-405c-955b-2a2794bf68fb?year=2023

29. Cloud platforms [Internet]. [cited 2024 Jan 25]. Available from: https://sis-tuni.funidata.fi/student/structure/otm-93167ecb-f4c2-45fb-821e-6f0588382df3

30. DevOps on AWS | Coursera [Internet]. [cited 2024 Mar 13]. Available from: https://www.coursera.org/specializations/aws-devops?utm_medium=sem&utm_source=gg&utm_campaign=B2C_EMEA__coursera_FTCOF_career-academy_pmax-multiple-audiences-country-multi&campaignid=20858198824&adgroupid=&device=c&keyword=&matchtype=&network=x&devicemodel=&adposition=&creativeid=&hide_mobile_promo&gad_source=1&gclid=Cj0KCQjw-r-vBhC-ARIsAGgUO2A4idVKYZkvzzJOR_eTGWvogwKYAdWQd99EzI0OdncUzGfRfMhhVtUaAggmEALw_wcB

31. Coursera [Internet]. [cited 2023 Oct 20]. Introduction to DevOps. Available from: https://www.coursera.org/learn/intro-to-devops

32. Coursera [Internet]. [cited 2023 Oct 20]. DevOps, Cloud, and Agile Foundations. Available from: https://www.coursera.org/specializations/devops-cloud-and-agile-foundations

33. Coursera [Internet]. [cited 2023 Oct 20]. IBM DevOps and Software Engineering. Available from: https://www.coursera.org/professional-certificates/devops-and-software-engineering?

34. Senior SW Developer - Tietoevry Care Data & Analytics | Tietoevry | LinkedIn [Internet]. [cited 2023 Oct 20]. Available from: https://www.linkedin.com/jobs/view/3703623024/?eBP=CwEAAAGKZPeQ9XMywyvKCg2voUAF6FKH7Wvp8VS9osCoOeRyG4RDfLRUV-g-jc_Uv07NF7QeEJKoLbEYVb8ByEg999ET9H4g6rAr50Y67OikcbYzW8ySmx_PDDOitfLJ4OMjUkTWigH2tl0tk9b5QN5m4iY6lkFB91hG9Ct8pwa_Gvncp0BBsEf9QlgvxJrQ8nQZ7sMTTjqrwbBz5OJVPMCPMmYQbmvLNeK1iTOYPYLaSPe8_xEls_JZ0C727Ttsy3grPEnSnr1Q-2d0TL3dp4RBk1V03DZlSB7zJMavJGY4SaUmhDlva9H3DAHZXPFhm31qZDZel22pLL4fQKYsskZxcLGojmTNFkNSTq9bs_kep81z725RJnZkz4gcfFAo1EyLS3uNpXo&refId=NtfmygYCJLKEkrPcp3SmUQ%3D%3D&trackingId=M8jI%2BpE5fvc%2BMdAcWl9ECw%3D%3D

35. DevOps Engineer | Top Remote Talent | LinkedIn [Internet]. [cited 2023 Oct 20]. Available from: https://www.linkedin.com/jobs/view/3677284043/?eBP=JOB_SEARCH_ORGANIC&refId=NtfmygYCJLKEkrPcp3SmUQ%3D%3D&trackingId=cHzALMpVTHAkMwzdAaDDWA%3D%3D

36. Data Platform Engineer [Internet]. [cited 2023 Oct 20]. Available from: https://www.linkedin.com/jobs/view/3697175796/?eBP=JOB_SEARCH_ORGANIC&r

efId=NtfmygYCJLKEkrPcp3SmUQ%3D%3D&trackingId=PkAf6IEK96vkCEZgrx0IRg%3D%3D

37. Cloud Support Engineer [Internet]. [cited 2023 Oct 20]. Available from: https://www.linkedin.com/jobs/view/3743402851/?alternateChannel=search&refId=e3db9dd9-31a7-427d-8b51-f1993df7c6ce&trackingId=goCL6R14Q2Wsbx41byK1gA%3D%3D

38. DevOps Engineers | KPMG Finland | LinkedIn [Internet]. [cited 2023 Oct 20]. Available from: https://www.linkedin.com/jobs/view/3709964626/?eBP=JOB_SEARCH_ORGANIC&refId=NtfmygYCJLKEkrPcp3SmUQ%3D%3D&trackingId=mzuuh%2FD06OIl6gaEcaRWPw%3D%3D

39. Google Forms: Online Form Creator | Google Workspace [Internet]. [cited 2023 Oct 31]. Available from: https://www.facebook.com/GoogleDocs/

40. SurveyMonkey [Internet]. [cited 2024 Mar 13]. Course Evaluation Survey Template & Questions. Available from: https://www.surveymonkey.com/templates/course-evaluation-survey-template/

41. Kiv AE, Soloviev VN, Semerikov SO. CTE 2018 – How cloud technologies continues to transform education. Available from: http://ds.knu.edu.ua/jspui/bitstream/123456789/992/1/Kiv%20A.%20E.%20CTE%202018%20%E2%80%93%20How%20cloud%20technologies.pdf

42. Valko NV, Kushnir NO, Osadchyi VV. Cloud technologies for STEM education. Available from: https://elibrary.kdpu.edu.ua/bitstream/123456789/3882/1/paper25.pdf

43. Vakaliuk TA, Spirin OM, Lobanchykova NM, Martseva LA, Novitska IV, Kontsedailo VV. Features of distance learning of cloud technologies for the organization educational process in quarantine. J Phys Conf Ser. 2021 Mar 1;1840(1):012051.

44. Qasem YAM, Abdullah R, Jusoh YY, Atan R, Asadi S. Cloud Computing Adoption in Higher Education Institutions: A Systematic Review. IEEE Access. 2019;7:63722–44.

45. Google Docs [Internet]. [cited 2023 Nov 3]. Learning Challenges in Container Technology Survey (Responses). Available from: https://docs.google.com/spreadsheets/d/141LL8sxBg8gjoJKxk1McSThGdTDgwZbizlfawXKyE7Y/edit?resourcekey&usp=embed_facebook