Online Learning Platform for Application-Inspired Cloud and DevOps Curriculum

Songjie Wang, Roshan Neupane, Ashish Pandey, Xiyao Cheng, Prasad Calyam
Department of Electrical Engineering and Computer Science, University of Missouri, MO 65211, USA
Email: wangso@missouri.edu; neupaner@missouri.edu; apfd6@mail.missouri.edu; xcheng@mail.missouri.edu; calyamp@missouri.edu

***Abstract*—Cloud-hosted services are being increasingly used in hosting business applications due to cost effectiveness, scalability, and ease of deployment. To facilitate rapid development, change and release process of cloud-hosted applications, the area of Development and Operations (DevOps) is fast evolving. It is necessary to train the future generation of application development professionals such that they are knowledgeable in the continuous integration and continuous delivery automation. In this paper, we present the design and development of our "Mizzou Cloud DevOps platform", an online platform to learn cutting-edge Cloud and DevOps tools/technologies using open/public cloud infrastructures for wide adoption amongst instructors/students. Our learning platform features scalability, flexibility, and extendability in providing Cloud and DevOps concepts knowledge and hands-on skills. We detail our "application-inspired learning" methodology that is based on integration of real-world application use cases in four learning modules that include laboratory exercises and self-study activities using realistic cloud/HPC-based application testbeds. We describe the learning modules that allow students to gain skills in using latest technologies (e.g., containerization, cluster and edge computing, data pipeline automation) to implement relevant security, monitoring, and adaptation mechanisms. Lastly, we evaluate our platform via a knowledge growth study survey with instructors and students.**

***Index Terms*—application-inspired learning, microservices, containerized workloads, monitoring/management of infrastructure, clusters and apps across multiple clouds**

I. Introduction

The term DevOps has been around for almost a decade and was originally created to combine popular practices related to software development (Dev) and IT operations (Ops) [1]. There are numerous definitions of the DevOps concept, but essentially its purpose is to automate and integrate the Dev and Ops processes for agile software development and operation teams, so that they can jointly develop, test, and release software faster and more reliably. The ultimate goal for DevOps is to shorten the application development life cycle while making sure the highest quality of software is accessible (with minimal defects) to the users through integration, testing, maintenance and delivery automation. In addition, DevOps is a mixture of essential components, including human, culture, platform, and tools that all work together to reduce friction in the processes between software development and release/operations.

This project is supported by VMware. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of VMware.

With the advances in Cloud and DevOps tools/technologies occurring rapidly, there is an ever-growing need for increasing their adoption in cloud-hosted application development processes. As a result, there is an urgent need to train the next-generation of application developers who can truly benefit in the software development achieved through Cloud and DevOps tools/technologies. Students need to have the relevant knowledge and skills to practice different concepts of Cloud and DevOps for effectively and efficiently developing applications. For the purpose of training skilled professionals in using Cloud and DevOps tools/technologies, there is a real need for effective online learning platforms for students to learn and practice Cloud and DevOps concepts and gain hands-on skills.

With Cloud and DevOps skills being in high demand for current and future software developers/engineers, there are currently multiple venues for students that include: industry vendor-specific certifications, Massive Open Online Courses (MOOCs), and course curriculum within education institutions [2]. More specifically, many public cloud infrastructure providers provide DevOps training and certification tracks such as e.g., DevOps and AWS [3], VMware Tanzu DevOps [4], QwikLabs DevOps essentials [5], Azure DevOps [6]. MOOCs offer a rich variety of DevOps training courses through popular platforms that include: LinkedIn Learning [7], Udemy [8], Udacity [9], Coursera [10], and Edx [11]. In addition, there are several U.S. institutions that have recently started offering Cloud and DevOps courses e.g., Harvard University [12], University of Minnesota [13], and John's Hopkins University [14].

However, we found that these learning venues often tend to require either lengthy time commitments and/or are costly for students. They are not ideal for students who are looking for open and sometimes free (no-cost) access to content and infrastructure to learn specific skills in a short period of time. Moreover, existing learning venues feature simplistic application use cases for students to extend their concepts learning to real-world applications. We found that there is a need for a learning platform for Cloud and DevOps concepts and hands-on skills building that is 'application-inspired' and thus helps students to leverage the tools/techniques in real-world application research and development projects. In addition, there is a dearth of an online learning platform that: (i) provides easy and open access to learning materials about different Cloud and DevOps tools/technologies, (ii) provides

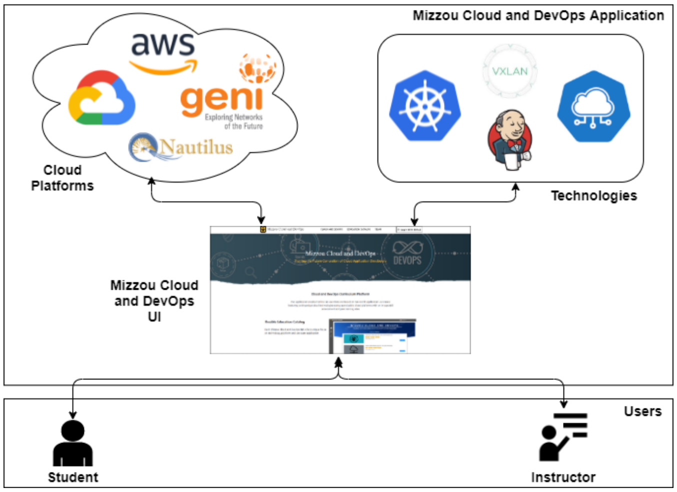

Fig. 1: Illustration of MCD platform with a user interface that allows students and instructors to access learning modules based on various cloud platforms and trending technologies.

the ability and flexibility to use both public (e.g., AWS, GCP) and research cloud platforms (e.g., Nautilus, GENI) in context of cutting-edge real-world applications use cases in learning modules, (iii) enables students to obtain peer ranking feedback, and (iv) enables instructors to effectively manage learning material and students' learning outcomes.

In this paper, we present the design and development of the "Mizzou Cloud and DevOps" (MCD) platform, an online platform for students to learn Cloud and DevOps related concepts, tools and technologies. Fig. 1 shows the MCD platform overview along with its components. The MCD platform offers free and easy access to learning modules with hands-on exercises that leverage different cloud infrastructures i.e., public clouds such as Amazon AWS [15], Google GCP [16], and large research cloud platforms such as PRP Nautilus [17] (Kubernetes cluster resources) and GENI [18] (at-scale networking resources). In addition, the MCD platform features an "application-inspired learning" methodology, which teaches students about popular Cloud and DevOps concepts based on integration of real-world application use cases. It also features a flexible and expandable catalog of four learning modules, and provides scalability through support for any number of students on the available cloud infrastructures. Further, it allows students to obtain rewards and rankings through a peer standing dashboard, and provides telemetry for instructors to monitor and manage students' learning outcomes.

The MCD platform's learning modules provide students with exposure to hands-on exercises and self-study activities using realistic cloud-based application testbeds. Using the "application-inspred learning" methodology, we have developed four hands-on learning modules using popular DevOps technologies related to research-based application use cases. These four learning modules available in the MCD platform include: (i) Module-1—Kubernetes [19] with Visual Cloud Computing App: this lab guides students to deploy and manage container-based application at-scale using clusters; (ii) Module-2—Software Defined Networking (SDN) with Cyber Defense App: this lab guides students to create virtual software-defined networks in Docker [20] containers using the Open vSwitch [21]; (iii) Module-3—KubeEdge [22] for edge computing with a Bioinformatics App: this lab guides students to create a KubeEdge cluster, and use the cluster to deploy a bioinformatics workflow on distributed edge computing resources; and (iv) Module-4—CI/CD with Jenkins with data-intensive Healthcare Web Apps: this lab teaches the student through use of the Jenkins [23] tool for Continuous Integration/Continuous Delivery in the process of agile software engineering. Throughout all these learning modules, students are expected to learn a variety of Cloud and DevOps tools/technologies, including but not limited to: containerization, trusted computing with HPC workflows, data pipeline automation, microservices, containerized workloads, monitoring/management of infrastructure, clusters and apps across multiple clouds. Given that Cloud and DevOps technologies are rapidly evolving, we use Continuous Integration/Continuous Delivery (CI/CD) best practices to regularly update our lab modules, create stable operating environments, and enable faster delivery of features of our online learning platform. To facilitate continuous learning and expansion of our MCD platform, we are creating additional learning modules featuring other DevOps tools/technologies e.g., continuous monitoring and testing, cluster orchestration, cloud security, and infrastructure as code (IaC) automation.

As part of our community outreach and online learning support, we are hosting community workshops to synchronously guide users of the MCD platform. In addition, we have created a Slack channel to offer asynchronous support to users looking for help in logistics and platform/technology issues. In our first series of community workshops, we organized a knowledge growth study survey involving 'before and after performing the lab modules' data collection with 45 participants. Our evaluation results show that there is significant growth in knowledge acquired by the participants and the benefits of the application-inspired learning of Cloud and DevOps technologies is highly agree-able to the participants.

The rest of the paper is organized as follows. In Section II, we describe the MCD platform design and development. In Section III, we detail the Cloud and DevOps learning modules in the MCD platform. Section IV discusses the evaluation of the MCD platform and Section V concludes the paper.

II. MCD Platform Design and Development

In this section, we first list the technical requirements and expected learning objectives for our MCD platform. Following this, we describe the various MCD platform features. Lastly, we detail the MCD platform components and their overall workflow in the lab exercises.

A. Technical Requirements and Learning Objectives

To effectively teach students about Cloud and DevOps, we carefully designed our learning modules to leverage a variety of Cloud and DevOps tools/technologies on public and research cloud infrastructures. The audience of our learning modules include cyberinfrastructure engineers, instructors, and online student learners. All the lab modules are independent and do not require students to follow any particular order to make progress. The salient technical requirements of our MCD platform are as follows:

- *Cloud Infrastructures*: We use public clouds such as Amazon Web Services and Google Cloud Platform to provide customizable and scalable cloud computing and networking capabilities. In addition, we use research clouds such as the Pacific Research Platform (PRP) Nautilus to provide easy to access Kubernetes clusters with hundreds of available nodes, and GENI to provide a fully fledged experimental network testbed for development and deployment of software-defined network applications.
- *Tools*: We use Docker to provide easy-to-work-on virtual nodes for use in most of the lab exercises. In this context, we use DockerHub [24] to provide a free-to-use cloud based container repository and also use GitHub [25] to provide user authentication and serve as a code repository.
- *Skill Set Background*: We expect the students using the learning labs in the MCD platform to have pre-requisite knowledge in basic computer networking, Docker container management, Linux scripting, and Python scripting. Students are provided with background reading materials for ready reference about technologies in a lab module.

The expected application-inspired learning objectives of the laboratory modules in the MCD platform for students are:

- Become competent to use the cloud platforms specified in the learning labs;
- Become familiar with setting up the required lab environment on the cloud platforms;
- Become competent in installing the required DevOps tools in the cloud;
- Become competent with basic configurations of the DevOps tools for a specific application;
- Become competent with the DevOps tool operations in the labs;
- Become competent with the advanced research-based use case applications in the labs;
- Become comfortable applying the tools/techniques learned from the labs and creating new applications/pipelines.

B. The MCD Platform Features

The MCD platform is designed to be versatile in supporting students' learning process and creation of new knowledge for various cloud services. It has the following features to maximize its impact on hands-on training for delivery knowledge of concepts and skill building of students:

1) Unique Education Catalog: MCD platform offers a flexible education catalog of four learning modules featuring learning of cutting-edge technologies using open/public cloud infrastructures that are easily accessible for wide adoption. Each module in the education catalog offers a unique focus on a particular technology, cloud infrastructure and use case application. For details on the lab modules, please refer to Section III.

2) Extendable Course Content: MCD platform uses a web services design and implementation process, which allows instructors and administrators to easily expand on the course content through easy integration of new training elements. The user interface has a dedicated section that allows these users to create new modules through a "Course Editor". The Course Editor not only allows instructors to view, update, and remove existing modules, but also facilitates creation of new modules out-of-the-box with no need for coding. The portal offers a web form that provides instructions to guide instructors to provide relevant materials in creating new lab modules, such as e.g., uploading of a lab manual, adding HTML pages, and providing assessment questions for student evaluation.

3) Real-world Application Use Cases: An essential aspect of hands-on learning in the MCD platform is to facilitate a learner with real-world problems that enable the learner to apply their learning to realistic application research and development. Towards this end, our MCD platform is supported by infrastructure configurations that are based on real-world application scenarios that include: Visual Cloud Computing (VCC), cyber defense, bioinformatics, and healthcare (COVID-19 related) Apps. Students can perform laboratory exercises using the infrastructure configurations facilitated by tutorials to explore, execute and evaluate various mechanisms on application infrastructures using cutting-edge technologies.

4) Easy Setup and Information Access: The MCD platform allows for easy-to-deploy cloud networks and DevOps practice scenarios by implementing a user-friendly interface with relevant instruction manuals. The manuals provide self-study steps to obtain in-depth understanding of various Cloud and DevOps tools/technologies involved in the exercises with relevant and quickly accessible documentation.

5) Scalable and On-demand: The learning modules are based on cloud infrastructures such as AWS, GCP, Nautilus and GENI which are capable of spinning up on-demand infrastructure instances for various applications. The resource utilization can be easily scaled from launching of minimal infrastructure for a single student to large infrastructure resources for tens/hundreds of students at the same time. The modular design of the MCD platform allows portability or integration with various cloud infrastructures, and the lab exercises can be deployed on any public cloud or research cloud infrastructure due to the use of Docker containers.

6) Ranking Dashboard and Peers Standing: The MCD platform consists of a comprehensive built-in evaluation mechanism based on ranked assessments to evaluate and monitor the learning progress and accomplishments of students. A gamification feature in the web portal allows students to view not only medals (i.e., Gold, Silver and Bronze) that signify their learning levels, but also their performance standing compared to their peers for different learning modules.

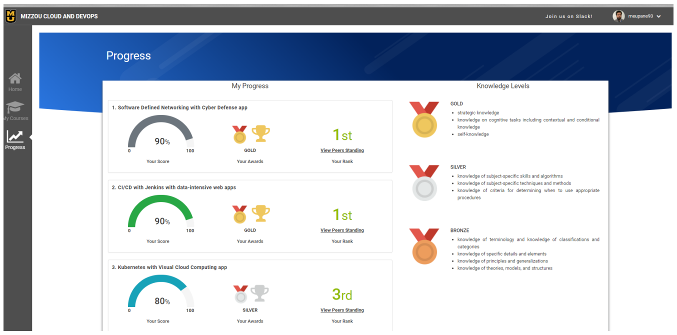

Fig. 2: MCD platform user interface showing the 'Progress' page of a student, with awards, score and rank along with the information of three knowledge levels, including Gold, Silver and Bronze levels.

7) Telemetry for Instructors: The MCD platform also allows instructors to view progress of students based on the data that is recorded throughout the student interactions with the learning modules. Interactive graphical representation of timeline of various activities within the portal such as e.g., student ranking, number of enrollments, score statistics can be visualized to derive conclusions on student accomplishments and refine the course content to enhance student learning experiences.

C. MCD Platform Components

The MCD platform includes several supporting components that are related to the lab exercises in order to concurrently deliver the learning modules for multiple students. The components also allow for realistic application scenarios to be launched in the cloud infrastructure through easy-to-follow instruction manuals that guide students through all the steps involved in the process of completing the lab exercises.

1) Web Portal: The MCD platform web portal is built as a central resource hub to provide a rich user experience to students and instructors. The user-friendly portal is designed with responsive views, simple navigation, contrasting color schemes and effective visualizations. It includes essential web pages to facilitate functionalities for both students and instructors, such as the education catalog, technology briefing section on tools and concepts used in the modules, wiki-based step-by-step user manual for the lab exercises, technical assessment user interface, and ranking dashboard with student peer standing.

2) Evaluation API server: Evaluation APIs are implemented to submit technical assessments acquired from students, to grade the tests automatically, and assign a knowledge level medal to individual students. We have developed an auto-grader as a part of the *submitEvaluation* API, which takes in key-value pairs of answers that have variant weightage for questions based on their complexity to calculate scores for students. Students will interface multiple choice questions for various lab exercises and their answers are auto-graded. After grading, *submitEvaluation* API assigns knowledge levels, as well as corresponding medals and peer rankings to students. Students can access their rankings, and medals, as shown in Fig. 2, as well as see other peer students' ranking through the *getStudentRank* API. The medals are assigned to students based on "Knowledge Dimension" concepts from Bloom's Taxonomy [26] with weights given to questions in increasing order for each level. The REST APIs are developed with *Spring Boot* [27], which is the most commonly used Java Web Framework.

3) Lab Exercise Workflow: Fig. 3 shows the sequence of actions of both students and instructors in the MCD platform. A student uses his/her Github account to login and needs to be approved by the instructor to be able to enroll into a lab exercise available in the education catalog to start the learning steps. Once approved and before starting the lab, the student is asked to submit a pre-lab survey to evaluate his/her initial knowledge/skill level. Each lab exercise has a section as part of the required reading which introduces the students with relevant references, tutorials on concepts and overview of technologies used in the learning modules to help with pre-requisites. From the web portal, students are provided with instruction manuals for performing the lab exercises that are tailored according to real-application scenarios such as container orchestration for video processing, cloud networking for cyber defense, edge/cloud federation for trusted computing, and data pipeline automation for knowledge discovery applications. The web portal has a wiki-based content containing essential instructions and guidelines for students to practice the lab exercises. The students gain experience in using various cloud platforms such as AWS, GCP, PRP Nautilus, GENI as they improve their understanding in making use of these infrastructures for application development.

After completing the lab exercise steps, the students need to complete a technical assessment associated with the corresponding lab exercise. The assessment aims to collect students' understanding of the concepts, tools and techniques involved at both the general and advanced levels. The assessment also quizzes students for their understanding of the different scripts that are used within the lab exercise. Finally, our implemented auto-grader grades the submitted answers and assigns medals and peer rankings to the students as illustrated in Fig. 2. A medal represents the final updated knowledge level of a student. The different knowledge levels are categorized as follows:

- Bronze (Level 1): We combined the first two categories of the knowledge dimension, the factual and the conceptual knowledge, to form the first level, the Bronze level. At this level, students have knowledge of the terminologies, classifications and categories, as well as some of the principles, theories, of the Cloud and DevOps concepts.
- Silver (Level 2): This level represents the procedural category of knowledge dimension. Student has knowledge of subject-specific skills, algorithms, and certain level of techniques, and has knowledge about when to use appropriate procedures in application development.
- Gold (Level 3): This level refers to the metacognitive category, which is strategic knowledge. At this level, the

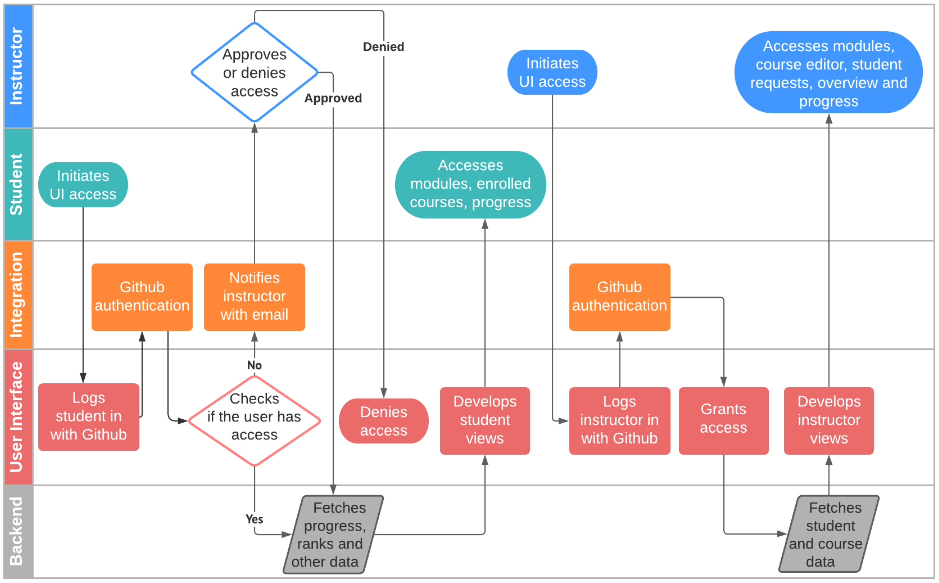

Fig. 3: MCD platform sequence flow for Student and Instructor user roles.

student masters concepts and skills required for building applications using relevant Cloud and DevOps concepts.

III. Cloud and DevOps Learning Modules

The MCD platform learning modules are designed with the purpose of understanding the fundamentals of popular Cloud and DevOps tools/technologies such as Kubernetes, KubeEdge, Jenkins, and SDNs. Students learn these concepts based on real-world application scenarios that involve automation and management, while also gaining experience on how real-world problems are solved. The learning modules are largely based on prior research outcomes in the areas of video processing [28], cyber defense using pretense [29], cloud federation for trusted computing [30], and data pipeline automation for knowledge discovery [31]. These various cloud based technologies are explored in the MCD platform such that the students can experience state-of-the-art Cloud and DevOps challenges.

A. Learning Modules Overview

The MCD platform learning modules are organized as four laboratory exercises and self-study activities to foster effective learning of Cloud and DevOps concepts in various in cloud infrastructures. The manuals and related documentation for the lab exercises are openly available online at [32]. While following these lab exercise manuals, students first read the related materials to learn about the basic concepts of specific technologies. Students then execute through the guided instruction manuals to gain hands-on experiences on the specific tools/technologies. Upon completion of the lab exercise steps, students are required to complete post-lab technical assessment to evaluate their skill and knowledge gained from a given lab exercise. Following the technical assessment, students are provided with medals, peer rankings and have visibility of their standing amongst their peers.

All the lab exercises represent both new and existing capabilities on open/public cloud platforms, and go further to help students learn about their detailed integration that is useful for a broad set of application contexts. The currently available lab modules in the MCD platform include: Module-1) Kubernetes with Visual Cloud Computing App; Module-2) Software Defined Networking with Cyber Defense App; Module-3) KubeEdge for Edge Computing with Bioinformatics App; and Module-4) CI/CD with Jenkins with Data-intensive Healthcare Web App.

All the learning modules on the MCD platform address a unique focus relating to a specific technology, platform, and application use case. In this following, we will detail the four learning modules in the MCD platform:

B. Module-1—Kubernetes with Visual Cloud Computing App

1) Technology: This module focuses on Kubernetes [19], a popular open-source system for automated deployments, scaling and management of containerized applications. Kubernetes offers many features for application development and operation, including configurability, scalability, flexibility, customizability, elasticity, and portability.

2) Platform: This module is available in PRP Nautilus [17] and AWS EKS [33]. PRP Nautilus is an openly accessible Kubernetes cluster supported by the National Science Foundation. PRP Nautilus runs up-to-date versions of the open-source Kubernetes software, so that students can use all the existing plugins and tools from the Kubernetes community. AWS EKS is a service for starting, running and scaling Kubernetes deployments.

3) Application Use Case: The application used in this module is a Visual Cloud Computing (VCC) application [28]. This application uses an image processing pipeline for object tracking in video streams. It follows a server/client architecture that is deployed on the Kubernetes cluster, and leverages deep learning algorithms using the Darknet with YOLO/COCO framework [34].

In this module, we guide the students to get familiar with the Nautilus Kubernetes cluster platform. They are provided with instructions to go through the fundamentals of Kubernetes, e.g., Pod, Deployment, Service, Load Balancing, Ingress, and Persistent Volumes. The students will deploy an example stateless application, the Guest Book application with MongoDB, to show how one can deploy an application without persistent data storage. The students will also deploy the VCC pipeline, and learn how to stream video data from Client pod, and analyze video frames with object tracking algorithms on the Server pod. They will then use a persistent volume to store results data and share them between the Client/Server.

C. Module-2—Software Defined Networking with Cyber Defense App

1) Technology: This module focuses on Software Defined Networking (SDN) using Virtual Extensible LAN (VXLAN). SDN is an approach to network management that enables dynamic, programmatically efficient network configuration. It is able to separate control plane from the data plane allowing a network to be programmed independent of the individual switches. We explore using VXLAN to enable SDN on Google Cloud Platform, allowing us to manage network configuration and routing, which can be achieved with SDN since cloud service providers typically do not allow users to manipulate their network switch hardware.

2) Platform: The cloud platforms we support for this module include both GCP and AWS. These cloud platforms allow us to integrate the application used on this module with ease of experimentation for various cyber defense by pretense scenarios and techniques.

3) Application Use Case: In this module, we use Dolus cyber defense by pretense system that was designed to perform attack detection and defense for various targeted attacks, such as distributed denial of service (DDoS), advanced persistent threats (APT), and Cryptojacking [29]. Dolus uses cyber defense using pretense principles to mitigate such targeted attacks. Using an SDN-based networking mechanism, this application allows re-direction of attacker's network connections into the system, which helps in thwarting targeted attacks through active defense.

In this lab module, students will learn how to setup SDN capability using the VXLAN protocol. They will create an SDN virtual network with a Pox Controller node as the control plane, and create two Open vSwitches, a Root Switch and a Slave Switch, which are connected to the Controller via a TCP/IP connection. The students then create a number of host VM instances that are connected to the two vSwitches via a VXLAN network overlay. These VM nodes work with various roles in the system. Some roles include benign users, attackers and controller. The SDN topology is used to allow experimentation with various Dolus cyber defense by pretense techniques on the cloud network infrastructure.

D. Module-3—KubeEdge for Edge Computing with Bioinformatics App

1) Technology: This module focuses on KubeEdge [22], an open-source system for extending native containerized application orchestration capabilities to hosts at the edge. It is built upon Kubernetes and provides fundamental infrastructure support for network, app deployment and metadata synchronization between cloud and edge computations.

2) Platform: This module is supported on GENI and AWS cloud platforms. GENI cloud is an openly accessible infrastructure supporting "at scale" research in networking, distributed systems, security, and novel applications. GENI is supported by the NSF and is available without charge (free) for research and classroom use. GENI offers various computing resource configurations from many locations around the United States, thus enabling distributed computing experimentation capabilities. AWS Educate accounts are also supported to perform this lab using KubeEdge.

3) Application Use Case: The application use case in this module is the volunteer edge computing for bioinformatics application. This application enables researchers to work with multi-cloud capabilities to better utilize compute resources available at local servers as well resources allocations on public cloud platforms. It gives researchers flexibility in creating multi-cloud architectures for their data-intensive and computation-intensive application requirements. Through a multi-cloud resource broker named OnTimeURB [30], this application allows the ideal compute resource allocations and stages the tasks on those resources during application execution.

In this lab, students will deploy a KubeEdge cluster on either the GENI cloud or on AWS. They will manage the cluster to include a Cloud Core node and multiple Edge Core nodes with various computing resources. Docker containers are then deployed with trusted bioinformatics workflows running HPC jobs based on various configurations. Volunteer edge computing algorithms are utilized to automatically select edge nodes to maximize performance while lowering related costs.

E. Module-4—CI/CD with Jenkins with Data-intensive Healthcare Web App

1) Technology: The technology students will explore in this lab is the Continuous Integration/Continuous Delivery (CI/CD) using Jenkins. Traditional integration comes with development practice that requires developers to integrate code into a shared repository several times a day, each of which is checked and then verified by build teams to detect problems early. This can be very tedious. Instead, a CI/CD system usually involves a tool that keeps a mandatory version control system. Whenever a change to version control system is detected, the system would automatically build and test the application, and deploy the newly built application on the server machine. This improves the change control in application through automation. In this module, we use Jenkins source automation server tool. Jenkins is the leading open-source automation server that provides hundreds of plugins to support building, deploying and automating any project application. It is a self-contained, open-source automation server which can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software. Jenkins can be installed through native system packages, Docker, or even run standalone on any machine with a Java Runtime Environment.

2) Platform: In this module, we use the AWS cloud infrastructure to create and configure a Jenkins server, and use the server to deploy example CI/CD applications and related data processing pipelines.

3) Application Use Case: This module uses the KnowCOVID-19 [31], an artificial intelligence (AI)-based publication analytics package that works through a guided web interface application. It is designed for COVID-19 clinical researchers to find helpful insights for decision-making on pandemic response by performing data mining of scientific publications. It enables filtering and ranking of data quality in the publications based on the evidence pyramid of scientific knowledge. The AI-based publication analytics is performed on published literature archives such as PubMed and LitCovid. This application also enables visualization of knowledge patterns to share with experts, drill down facts and study deep topic relationships.

In this module, the students will create and deploy a Jenkins server on the AWS cloud infrastructure. The instructions in the tutorials are explained such that the students get familiar with the basic functionalities of Jenkins. The KnowCOVID-19 application enables students in using simple application development increments (whenever model training is updated) to learn how to create and run CI/CD jobs on Jenkins.

IV. Evaluation

We organized a community workshop series for instructors and students from various universities in the United States to evaluate our MCD platform in terms of knowledge growth and workshop outcomes. In this section, we summarize the results obtained from 45 participants with diverse backgrounds in terms of age, technical knowledge and work experience.

A. Knowledge Growth Study

We performed a pre-lab evaluation survey and a post-lab technical assessment to evaluate knowledge growth of each participant. According to participants' answers, they were assigned an initial knowledge level, ranging from Level 1 through Level 3. We also assigned a Level 0 for the participants that have no knowledge of the platform or the technologies used on a learning module to form a four knowledge dimensions from Bloom's taxonomy. We then performed a comparative study to compare their knowledge level before and after the lab module completion. Some example pre-lab questions for the first lab include:

- Are you familiar with Kubernetes?
- Have you used PRP Nautilus cloud platform before?
- Have you worked with Docker containers before?

The post-lab technical assessment featured a set of multiple choice questions prepared based on the concepts used in e.g., Module-1 along with technical questions that include:

- In the context of Kubernetes, what is a container?
- Which of the following commands is used to create a Kubernetes service?
- Given a set of pods, how can one enable network access between them using Kubernetes?
- How are pods created?

Based on the survey responses, participants were assigned knowledge levels as part of a statistical comparative study as illustrated in Fig. 4. The results illustrated in pie chart show that 93% of the participants in Level 0 and Level 1 were upgraded to at least Level 2, which demonstrates a notable knowledge growth in the participants. Moreover, many participants who were at Level 2 or below could enhance their learning and move to Level 3 as we see a rise from 5% to 43%. This strongly demonstrates that the lab module completion certainly helped the participants to improve their skill level in the technologies explored in the Module-1.

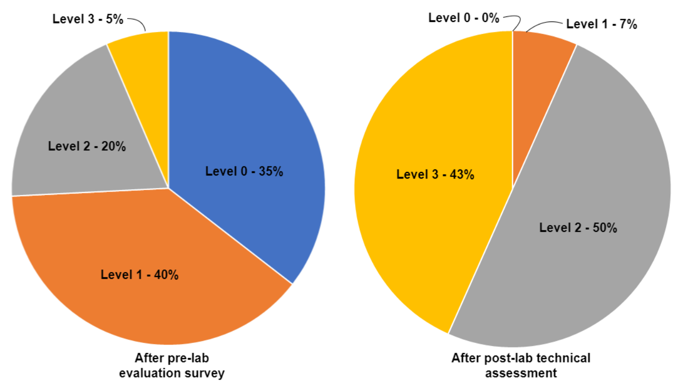

Fig. 4: Percentages of students in each knowledge level before (left) and after (right) completing Module-1 based on pre-lab and post lab assessments

B. Feedback on Workshop Outcomes

The participants at the workshop were provided with a feedback survey to collect their opinion and improvement suggestions for the MCD platform. In the following, we show a sample set of 5 questions in the survey questionnaire:

- Our approach to using application-inspired real-world case studies to learn cloud platforms and DevOps technology is helpful
- Our labs featuring support for multiple cloud platforms are useful
- The technologies related to micro-services and containerized workflow within clusters are relevant and timely
- The ranking dashboard is helpful for visualizing and assessing your learning outcome
- Overall I would rate my experience as successful

Using the Likert Scale (range involves 'Strongly Disagree' to 'Strongly Agree'), we obtained the survey results presented in Fig. 5. We concluded that our approach of using application-inspired real-world case studies to learn the Cloud and DevOps

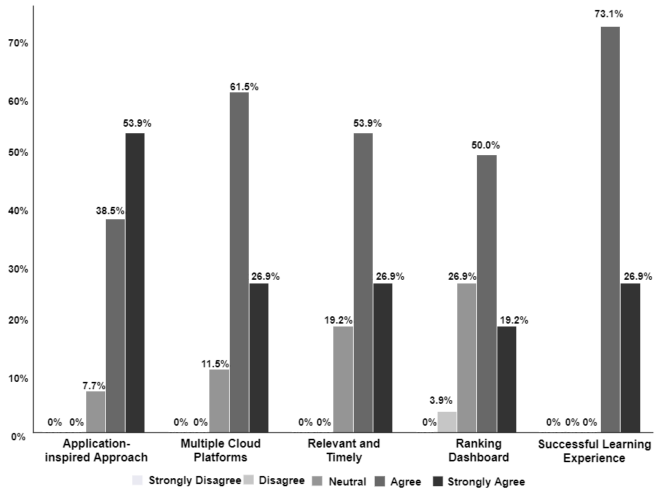

Fig. 5: Percentages of workshop attendees distributed with respect to their feedback on the questionnaire provided

technologies was highly agree-able to the participants. The attendees also seemed excited with exposure to the various public and research cloud platforms we are supporting in the MCD portal. Most of the attendees also agreed that the technologies and concepts we included in the learning modules are timely and relevant to cloud application development.

V. Conclusion

In this paper, we presented the MCD platform, an interactive and user-friendly platform for learning Cloud and DevOps concepts that involve use of state-of-the-art tools/technologies such as e.g., Kubernetes, KubeEdge, SDN using VXLAN, CI/CD with Jenkins. Four learning modules are currently being delivered via our MCD platform, and each module features lab exercises that are for relevant application use cases in container cluster orchestration, cyber defense, trusted edge computing, and data pipeline automation. We described our design principles, experiences and findings in the development of MCD platform to integrate desired student learning outcomes pertaining to various real-world applications. The MCD platform can be applied in Cloud and DevOps curriculum for learning workflow automation and change management via hands-on modules, and also in extra curricular activities pertaining to advanced cloud application development.

References

[1] "How DevOps can accelerate software development process," Accessed: August 25, 2021. Available at https://www.embeddedcomputing.com/technology/software-and-os/ides-application-programming/how-devops-can-accelerate-software-development-process.

[2] "How DevOps is transforming higher education, June 2021." Accessed: August 27, 2021. Available at https://news.elearninginside.com/how-devops-is-transforming-higher-education/.

[3] "DevOps and AWS." Accessed: August 27, 2021. Available at https://aws.amazon.com/devops/.

[4] "VMware Tanzu DevOps," Accessed: August 20, 2021. Available at https://tanzu.vmware.com/devops.

[5] "DevOps Essentials, by QwikLabs," Accessed: August 25, 2021. Available at https://www.qwiklabs.com/quests/96.

[6] "Azure DevOps," Accessed: August 20, 2021. Available at https://azure.microsoft.com/en-gb/services/devops.

[7] "LinkedIn Learning [online]," Accessed: August 03, 2021. Available at https://www.linkedin.com/learning.

[8] "Udemy [online]," Available at https://www.udemy.com.

[9] "Udacity [online]," Available at https://www.udacity.com.

[10] "Coursera [online]," Available at https://www.coursera.com.

[11] "Edx [online]," Available at https://www.edx.com.

[12] "An Introduction to Cloud and DevOps Concepts, Harvard University. [online]," Accessed: August 17, 2021. Available at https://scholar.harvard.edu/molloy/classes/introduction-cloud-and-devops-concepts.

[13] "DevOps II: Development Strategies, University of Minnesota. [online]," Accessed: August 17, 2021. Available at https://ccaps.umn.edu/credit-courses/devops-ii-development-strategies.

[14] "DevOps and Secure Software Development, John's Hopkins University. [online]," Accessed: August 17, 2021. Available at https://ep.jhu.edu/courses/605609-devops-and-secure-software-development/.

[15] "Amazon AWS Cloud. [online]," Accessed: August 25, 2021. Available at https://aws.amazon.com/.

[16] "Google Cloud Platform. [online]," Accessed: August 25, 2021. Available at https://cloud.google.com/.

[17] "PRP Nautilus Cloud. [online]," Accessed: August 25, 2021. Available at https://nautilus.optiputer.net/.

[18] "GENI - Open cloud infrastructure. [online]," Accessed: May 22, 2021. Available at https://portal.geni.net/.

[19] "Kubernetes. [online]," Accessed: July 13, 2021. Available at https://kubernetes.io/.

[20] D. Merkel, "Docker: lightweight linux containers for consistent development and deployment," *Linux journal*, vol. 2014, no. 239, p. 2, 2014.

[21] B. Pfaff, J. Pettit, T. Koponen, E. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar *et al.*, "The design and implementation of open vswitch," in *12th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 15)*, 2015, pp. 117–130.

[22] Y. Xiong, Y. Sun, L. Xing, and Y. Huang, "Extend cloud to edge with kubeedge," in *2018 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 2018, pp. 373–377.

[23] "Jenkins [online]," Accessed: July 20, 2021. Available at https://jenkins.io/.

[24] "DockerHub [online]," Accessed: July 21, 2021. Available at https://hub.docker.com/.

[25] "GitHub [online]," Accessed: July 21, 2021. Available at https://www.github.com/.

[26] B. S. Bloom *et al.*, "Taxonomy of educational objectives. vol. 1: Cognitive domain," *New York: McKay*, vol. 20, no. 24, p. 1, 1956.

[27] "Spring Boot [online]," Accessed: May 27, 2021. Available at https://spring.io/projects/spring-boot.

[28] C. Qu, P. Calyam, J. Yu, A. Vandanapu, O. Opeoluwa, K. Gao, S. Wang, R. Chastain, and K. Palaniappan, "DroneCOCoNet: Learning-based edge computation offloading and control networking for drone video analytics," *Future Generation Computer Systems*, vol. 125, pp. 247–262, 2021.

[29] R. L. Neupane, T. Neely, P. Calyam, N. Chettri, M. Vassell, and R. Durairajan, "Intelligent defense using pretense against targeted attacks in cloud platforms," *Future Generation Computer Systems*, vol. 93, pp. 609–626, 2019.

[30] A. Pandey, Z. Lyu, T. Joshi, and P. Calyam, "OnTimeURB: Multi-cloud resource brokering for bioinformatics workflows," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2019, pp. 466–473.

[31] R. Oruche, V. Gundlapalli, A. P. Biswal, P. Calyam, M. L. Alarcon, Y. Zhang, N. R. Bhamidipati, A. Malladi, and H. Regunath, "Evidence-based recommender system for a covid-19 publication analytics service," *IEEE Access*, 2021.

[32] "Mizzou Cloud and DevOps [Online]," Available at *https://tinyurl.com/48bzwuj4*.

[33] "Amazon Elastic Kubernetes System (EKS). [online]," Accessed: October 04, 2021. Available at https://aws.amazon.com/.

[34] R.-C. Chen *et al.*, "Automatic license plate recognition via sliding-window darknet-YOLO deep learning," *Image and Vision Computing*, vol. 87, pp. 47–56, 2019.