This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

Online Self-Service Learning Platform for Application-Inspired Cloud Development and Operations (DevOps) Curriculum

[1]Roshan Lal Neupane, [1]Prasad Calyam, [1]Songjie Wang, [1]Kiran Neupane, [1]Ashish Pandey, [1]Xiyao Cheng, [1]Durbek Gafurov, [1]Hemanth Sai Yeddulapalli, [2]Noah Glaser, [2]Kanu Priya Singh, [2]Yuanyuan Gu, [2]Shangman (Eunice) Li, [34]Sharan Srinivas

[1]Center for Cyber Education, Research and Infrastructure; [2]Information Experience Lab; [3]Department of Industrial and Systems Engineering, [4]Department of Marketing, University of Missouri; Email: {neupaner, calyamp, wangso, kngbq, hygw7, noah.glaser, srinivassh} @missouri.edu; {apfd6, xcheng, sli, durbek.gafurov}@mail.missouri.edu; kanupriya.singh@yale.edu; yggcc@umsystem.edu

***Abstract*—Cloud-hosted services are being increasingly used in hosting business and scientific 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 scientific application development professionals such that they are knowledgeable in the DevOps-enabled continuous integration/delivery automation. In this paper, we present the design and development of our "Mizzou Cloud DevOps platform", an online self-service platform to learn cutting-edge Cloud 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 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 eight learning modules that include laboratory exercises and self-study activities. The learning modules 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. The evaluation of our platform features a knowledge growth study to assess student learning, followed by a usability study to assess the online learning platform as well as the curriculum content as perceived by instructors and students across multiple hands-on workshops.**

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

I. Introduction

The term DevOps, coined nearly a decade ago, aimed to merge key practices in software development (Dev) and information technology operations (Ops) [1]. While definitions vary, its core objective is to automate and unify Dev and Ops processes for agile teams, enabling faster and more reliable software development, testing, and deployment. DevOps ultimately seeks to shorten the application development life cycle while ensuring high-quality, defect-minimized software accessibility to users through automation in integration, testing, maintenance, and delivery.

With the growth in digital transformation among many organizations, the DevOps practice is commonly performed through the usage of cloud computing resources. For example, about 80% of organizations are considering to use "Cloud DevOps" to transform their current manual and siloed processes used in evolving and scaling data/compute-intensive applications [2]. The overarching benefit of Cloud DevOps is that it enables essential components, including humans, culture, platform, and tools to work together in order to reduce friction in the processes between software/infrastructure development and release/operations. Given such a potential, the market size of Cloud DevOps is expected to grow from USD 7.4 billion in 2021 to USD 37.2 billion by 2030 at a Compound Annual Growth Rate (CAGR) of 20% [3].

Given the high demand for Cloud DevOps expertise among current and prospective software developers/engineers, there exist various avenues for students to acquire these skills. These include industry-specific certifications from vendors, Massive Open Online Courses (MOOCs), and educational institutions [4] integrating DevOps into their curricula. Notably, many public cloud providers offer dedicated DevOps training and certification programs, such as DevOps and AWS [5], VMware Tanzu DevOps [6], QwikLabs DevOps essentials [7], and Azure DevOps [8]. MOOC platforms like LinkedIn Learning [9], Udemy [10], Udacity [11], and Coursera [12] host a diverse range of DevOps training courses. Additionally, some U.S. institutions, including Harvard University [13], University of Minnesota [14], and John's Hopkins University [15], have recently introduced Cloud DevOps courses.

However, existing learning venues feature high costs for accessing infrastructure, and do not often feature comprehensive and holistic training materials, application-inspired learning, or self-assessment and peer feedback mechanisms. There's a critical demand for an online, self-service learning platform tailored to Cloud DevOps, offering: (i) freely and openly accessible learning resources covering essential tools and technologies, (ii) hands-on experience with real-world

This project is supported by the VMware University Research Fund. 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.

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

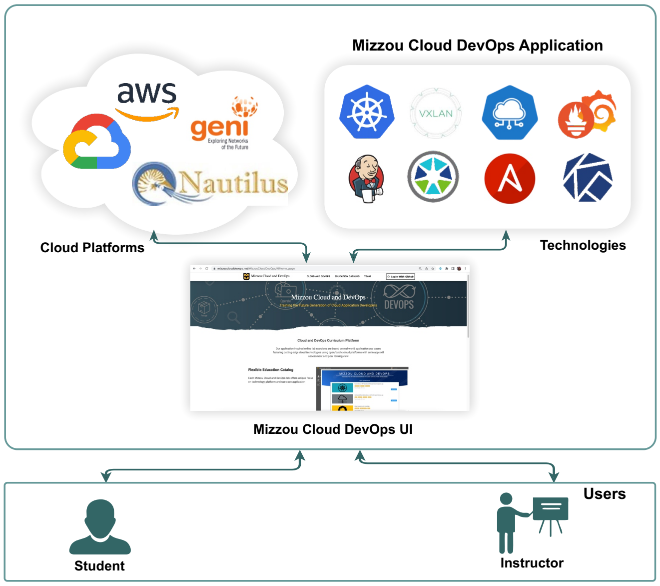


Fig. 1: The depiction of the MCD platform showcases a user interface enabling both students and instructors to navigate learning modules centered around diverse cloud platforms and emerging technologies.

'application-inspired' use cases, integrating relevant cloud resources, (iii) peer ranking feedback for students, and (iv) efficient learning material management and assessment for instructors. In this context, our application use-cases are focused on experimentation that take place for real-world application domains such as e.g., healthcare, cybersecurity, supply chain, etc., with real data generated from scientific workflows.

This paper outlines the design and creation process of the "Mizzou Cloud DevOps" (MCD) platform, an online self-service platform crafted to educate students on Cloud DevOps principles, tools, and technologies. Fig. 1 illustrates an overview of the MCD platform alongside its constituent elements. Providing complimentary access, the MCD platform furnishes learning modules equipped with practical exercises that capitalize on diverse cloud infrastructures, encompassing renowned public clouds like Amazon Web Services (AWS) [16] and Google Cloud Platform (GCP) [17], as well as expansive research cloud platforms such as PRP Nautilus [18] (utilizing Kubernetes cluster resources) and GENI [19] (a future Internet experimentation facility). Furthermore, the MCD platform employs an "application-inspired learning" approach, imparting knowledge of prevalent Cloud DevOps concepts like Kubernetes, VMWare Tanzu, KubeEdge, Jenkins, Ansible, Kubeflow, Prometheus, and Grafana through the integration of real-world application scenarios (e.g., cybersecurity, healthcare, video analytics). It features expandable collection of eight learning modules and is engineered to accommodate the training needs of any number of students.

The learning modules available in the MCD platform include: (i) Module-1—Kubernetes [20] with a Visual Cloud Computing application; (ii) Module-2—Software Defined Networking (SDN) with a Cyber Defense application; (iii) Module-3—KubeEdge [21] for edge computing with a Bioinformatics application; (iv) Module-4—CI/CD with Jenkins with a data-intensive Healthcare web application; (v) Module-5—Tanzu Community Edition [22] with a Drone Video Streaming and Analytics application; (vi) Module-6—Ansible [23] for Infrastructure as Code with Cybersecurity Operations in a Supply Chain/Logistics application; (vii) Module-7—Kubeflow [24] on AWS with a Health Informatics application; and (viii) Module-8—Prometheus [25] and Grafana [26] with a Video Transmission application.

Moreover, it enables students to earn rewards and rankings via a peer evaluation system, while offering instructors the ability to track and oversee students' progress in areas such as containerization, dependable high-performance computing workflows, automated data pipelines, microservices, containerized workloads, and the monitoring and management of infrastructure, clusters, and applications across various cloud platforms.

As part of our efforts to engage with the community and enhance the learning experience, we organized a series of community workshops aimed at guiding users of the MCD platform. Within these workshops, we conducted a survey to gauge knowledge growth, collecting data both before and after participants completed lab modules, with a total of over 200 participants. Our assessment revealed significant knowledge advancement among participants, indicating a favorable reception towards the application-focused learning approach in Cloud DevOps technologies. Additionally, we conducted a usability study with individuals from non-computer science backgrounds to explore ways to improve the accessibility of our learning modules to a broader audience. Finally, we conducted focus group sessions to gather qualitative feedback and assess the advantages of our MCD platform.

The rest of the paper is organized as follows. In Section I, we discuss the related works. In Section III, we describe the MCD platform design and development. In Section IV, we detail the Cloud DevOps learning modules in the MCD platform. Section V discusses the evaluation of the MCD platform. Section VI discusses implications and limitations of the study and Section VII concludes the paper.

II. Related Works

A. Cloud DevOps for Application Workflows

There have been prior efforts that have shown the benefits of Cloud DevOps in data/compute-intensive application workflows. Given that such workflows require expensive computing resources and custom workflow configurations, the work in [34] showed that users face challenges in lowering costs for development, and in the reproducibility of such workflows. Correspondingly, the work in [35] suggests the benefits in the use of Cloud DevOps concepts (e.g., configuration automation) for improving the reproducibility of data/compute-intensive workflows. It also highlights ideas such as parallel programming, and configuration automation through `YAML` files for better experience and results. Work in [36] explores the challenges and benefits of using DevOps concepts for improving software development in data-intensive application

TABLE I Online learning platforms and their features for teaching Cloud DevOps.

Features	**MCD**	**Open Platforms: [27]–[32]**	**Proprietary Platforms: [5]–[12], [33]**
Comprehensive and holistic training materials	✓	×	✓
Free of cost infrastructure access	✓	✓	×
Cutting-edge real-world application-inspired learning	✓	×	×
Self assessment and peer feedback mechanism	✓	×	×
Designed for students of both computer science and non-computer science background	✓	×	×

projects. Other works such as [37], [38] explored the advantages of using Cloud DevOps concepts for improving scientific software development projects in domains of information management, petroleum geology and geophysics.

Our novel work on application-inspired learning focuses on using Cloud DevOps for exemplar data/compute-intensive application workflows involving e.g., video transmission/processing, bioinformatics workflow management with edge computing, network security configuration and auditing, machine learning based model building for health informatics.

B. Online Learning Platforms for Cloud DevOps

A notable prior work on an online learning platform design for Cloud DevOps can be seen in [27], where Java developers learn skills for setting up a pipeline, testing, and deployment of a simple web application. Recommendations on software development curricula involving Cloud DevOps for students with computer science background are discussed in [28]–[30]. Work in [31] implements a cloud-based DevOps Lab Platform based on a Git-Lab code repository and a continuous integration framework. Authors in [32] discuss an online platform for teaching students DevOps that features monitoring and management of their practicing of relevant concepts, and teachers can evaluate students' work.

Although there are benefits (e.g., decreased time of development, more stable operations, increased reproducibility and reliability) of using Cloud DevOps for application workflow management, available online learning platforms do not cater to learners from both computer science, and non-computer science backgrounds. Importantly, there is currently a gap in the existing body of knowledge in existing online learning platforms for Cloud DevOps [39]. Specifically, open learning platforms as discussed in [27]–[32], and industry-based proprietary learning platforms [5]–[12], [33] are lacking in addressing one or more of the important features that are relevant for learners and teachers as shown in Table I.

The novelty of our proposed Mizzou Cloud DevOps platform is in the comprehensive development of a set of real-world application-inspired learning modules for online self-service learning to effectively/efficiently integrate Cloud DevOps within data/compute-intensive application workflows. Our MCD platform approach to use public cloud (e.g., AWS) and research cloud (e.g., Nautilus) resources, and supporting features such as self-assessment and peer feedback can elevate the learning experiences for students with, and without computer science backgrounds.

III. MCD Platform Design and Development

In this section, we begin by outlining the technical prerequisites necessary to fulfill the learning objectives within our MCD platform. Subsequently, we explain the diverse features offered by the MCD platform. Finally, we explain the components of the MCD platform and their flow within the exercises embedded in the learning modules.

A. Technical Requirements for Learning Objectives

In order to proficiently educate students on Cloud DevOps, we meticulously crafted our learning modules to utilize a diverse array of Cloud DevOps tools and technologies across both public and research cloud infrastructures. Our target audience encompasses cyberinfrastructure engineers, instructors, and online student learners. Each lab module stands alone and does not necessitate adherence to a specific sequence for advancement. The essential technical prerequisites for attaining the learning goals within our MCD platform are outlined as follows:

- *Cloud Infrastructures*: We employ renowned public cloud services like Amazon Web Services (AWS) [16] and Google Cloud Platform (GCP) [17] for scalable cloud computing and networking functionalities that can be personalized. Furthermore, we utilize research cloud platforms like the Pacific Research Platform (PRP) Nautilus [18], offering straightforward access to Kubernetes clusters with hundreds of nodes, and GENI [19], providing a comprehensive experimental network testbed for the development and implementation of software-defined network applications.
- *Tools*: For most of our lab exercises, we employ Docker to spawn user-friendly virtual nodes. Within this framework, we utilize DockerHub [40] as a complimentary cloud-based container repository and leverage GitHub [41] for user authentication and as a repository for code storage.
- *Skill Set Background*: Students engaging with the learning labs on the MCD platform are expected to possess basic knowledge in computer networking, Linux scripting, and Python scripting. Background reading materials are furnished to students within each learning module for convenient reference regarding the technologies covered. The students do not require to have background knowledge in the technologies taught via our learning modules, such as Kubernetes, KubeEdge, KubeFlow, Ansible, and SDN.

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

B. The MCD Platform Features

The MCD platform is crafted to be adaptable, aiding students in their learning journey and fostering the development of fresh insights across a spectrum of cloud services. It presents the following attributes to optimize its efficacy in providing practical student training for both conceptual understanding and skill enhancement:

1) Unique Education Catalog: The MCD platform provides a versatile educational catalog comprising eight learning modules that delve into the exploration of cutting-edge technologies (e.g., Kubernetes, KubeEdge, Prometheus, Jenkins, OpenFlow, Ansible, KubeFlow) utilizing open/public cloud infrastructures, which are readily accessible for broad implementation. Each module within the educational catalog concentrates on a specific technology, cloud infrastructure, and application use case. For further information on the learning modules, please consult Section IV.

2) Extendable Course Content: The MCD platform employs a web services approach to design and implementation, enabling instructors to seamlessly enrich course content by integrating new training components. Within the user interface, there's a dedicated section accessible to instructors known as the "Course Editor." This tool not only allows instructors to manage existing modules by viewing, updating, or removing them but also facilitates the creation of new modules effortlessly, without requiring coding expertise. The portal features a user-friendly web form that guides instructors through the process of creating new lab modules. This includes tasks such as uploading lab manuals, incorporating HTML pages, and supplying assessment questions for student evaluation.

3) Real-world Application Use Cases: A crucial element of experiential learning within the MCD platform is to empower learners to tackle real-world problems for educating. To achieve this goal, our platform is underpinned by infrastructure setups modeled after real-world application scenarios. These scenarios encompass tasks like video transmission/processing, bioinformatics workflow management with edge computing, network security configuration and auditing, and machine learning-based model building for health informatics. Through tutorials and facilitated infrastructure configurations, students engage in laboratory exercises to explore, execute, and assess diverse mechanisms on application infrastructures utilizing state-of-the-art technologies.

4) Easy Setup and Information Access: The MCD platform facilitates the setup of cloud networks and DevOps practice scenarios with ease, thanks to its user-friendly interface and comprehensive instruction manuals. These manuals offer step-by-step guidance for self-study, enabling users to gain a thorough understanding of the different Cloud DevOps tools and technologies utilized in the exercises, with readily available and pertinent documentation. As shown in Fig. 2, instructions in the lab manuals have been developed with Markdown [42], a lightweight markup language for creating formatted text using a plain-text editor. This allows learners to conveniently navigate through various learning chapters pertaining to the modules, and use a code block copy feature for reducing typographical errors during execution of commands on terminal sessions.

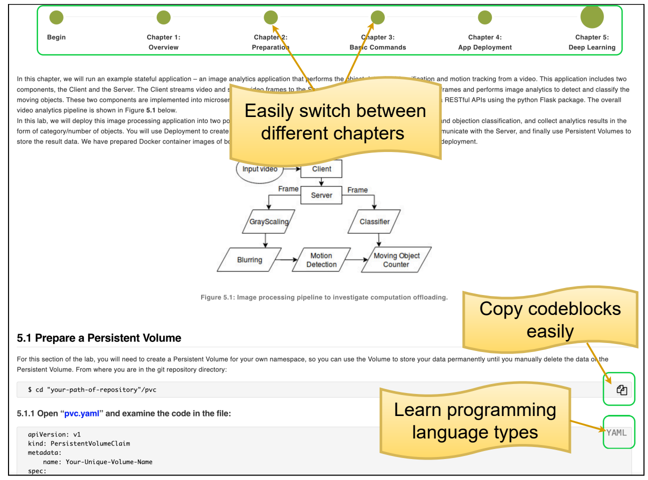


Fig. 2: MCD platform user interface layout for showing a chapter in the Module-1 with ability for students to e.g., switch different chapters of a module, copy codes from code blocks.

5) Scalable and On-demand: The learning modules utilize cloud infrastructures like AWS, GCP, Nautilus, and GENI, capable of swiftly provisioning infrastructure instances as needed for diverse applications. Resource allocation can seamlessly adjust, scaling from minimal infrastructure for individual students to expansive resources accommodating tens or hundreds of students concurrently. The MCD platform's modular design enables portability and integration with different cloud infrastructures, while lab exercises can be deployed on any public or research cloud infrastructure, facilitated by the use of Docker containers.

Fig. 3: 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.

6) Ranking Dashboard and Peers Standing: The MCD platform incorporates a thorough built-in evaluation system utilizing ranked assessments to assess and track students' learning progress and achievements. Illustrated in Fig. 3, a gamification element enables students to observe their learning levels represented by medals (Gold, Silver, and Bronze), as well as their performance ranking relative to peers across various learning modules.

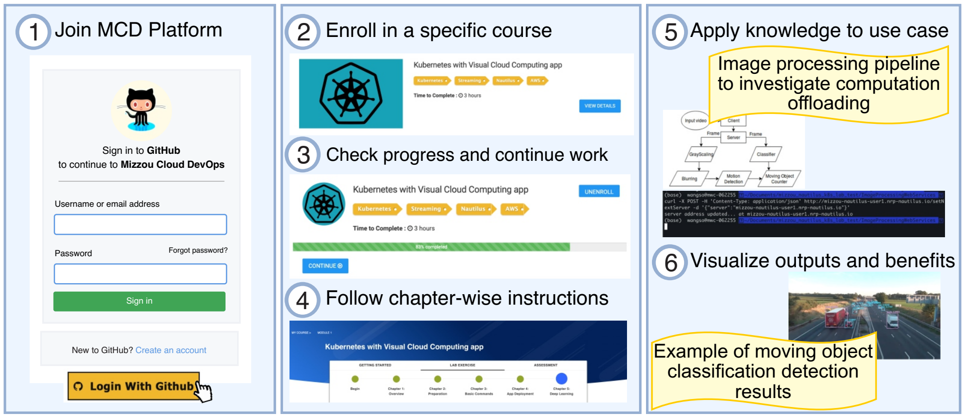


Fig. 4: Illustration of user steps in Mizzou Cloud DevOps platform.

7) Telemetry for Instructors: Additionally, the MCD platform enables instructors to monitor students' progress by analyzing the data collected during their interactions with the learning modules. Interactive graphical representations, like timelines displaying activities such as student rankings, enrollment numbers, and score statistics within the MCD web portal, offer insights into student achievements. These insights can then be used to refine course content, thereby enhancing the overall learning experience for students.

C. MCD Platform Components

The MCD platform comprises numerous supplementary elements associated with lab exercises, facilitating the simultaneous delivery of learning modules to multiple students. These components also enable the launch of realistic application scenarios in the cloud infrastructure, with user-friendly instruction manuals guiding students through each step of completing the lab exercises.

1) Web Portal: Fig. 4 shows an illustration of our front-end implementation (developed using AngularJS) that guides a user through a series of steps to complete a set of learning module exercises. We made the platform access easy/secure by requiring users to use their GitHub credentials to access various learning modules. The MCD web portal serves as a centralized resource hub, offering students and instructors a robust user experience. Designed with simplicity and accessibility in mind, the portal features responsive views, intuitive navigation, contrasting color schemes, and impactful visualizations. The user dashboard features two roles, i.e., instructor and learner. The instructor is able to manage the learning modules as *courses*, and can monitor student progress through a telemetry dashboard. Learners can enroll in a specific course, follow the hands-on lab exercises, and apply knowledge to a real-world application use case e.g., video analytics to visualize outputs and benefits. Moreover, the learner can take a technical assessment after completing the exercises to obtain a peer ranking score.

2) Learning Evaluation API: We implemented a set of Learning Evaluation APIs to obtain technical assessments from students, to grade the tests automatically, and assign a knowledge level medal to individual students. We've created an auto-grader within the *submitEvaluation* API, which processes answers provided by students in key-value pairs. These answers carry varying weights depending on the complexity of the questions, allowing for the calculation of scores. Students engage with multiple-choice questions across different lab exercises, and their responses are automatically evaluated. Following grading, the *submitEvaluation* API assigns knowledge levels along with corresponding medals and peer rankings to students. Students can access their rankings and medals, depicted in Fig.3, and view the rankings of their peers through the *getStudentRank* API. The medals are allocated to students based on the "Knowledge Dimension" concepts outlined in Bloom's Taxonomy [43], with question weights increasing sequentially for each level. These REST APIs are developed using *Spring Boot* [44], a popular Java Web Framework.

3) Lab Exercise Workflow: Figure 5 illustrates the sequence of actions performed by both students and instructors within the MCD platform. A student initiates the process by logging in using their Github account and must await approval from the instructor before enrolling in a learning module from the education catalog. Upon approval and prior to commencing the lab, the student is prompted to complete a pre-lab survey to assess their initial knowledge/skill level. Each learning module includes a required reading section that acquaints students with relevant references, tutorials, and an overview of the technologies employed in the modules, aiding in meeting prerequisites. The web portal furnishes students with instruction manuals tailored to real-world 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. Additionally, the web portal hosts a wiki-based content featuring essential instructions and guidelines for students to practice the lab exercises. As students progress, they gain proficiency in utilizing various cloud platforms, including AWS, GCP, PRP Nautilus, and GENI, thereby enhancing their understanding of leveraging these infrastructures for application development.

Upon finishing the lab exercise steps, students are required to undertake a technical assessment related to the respective learning module. This assessment is designed to gauge

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

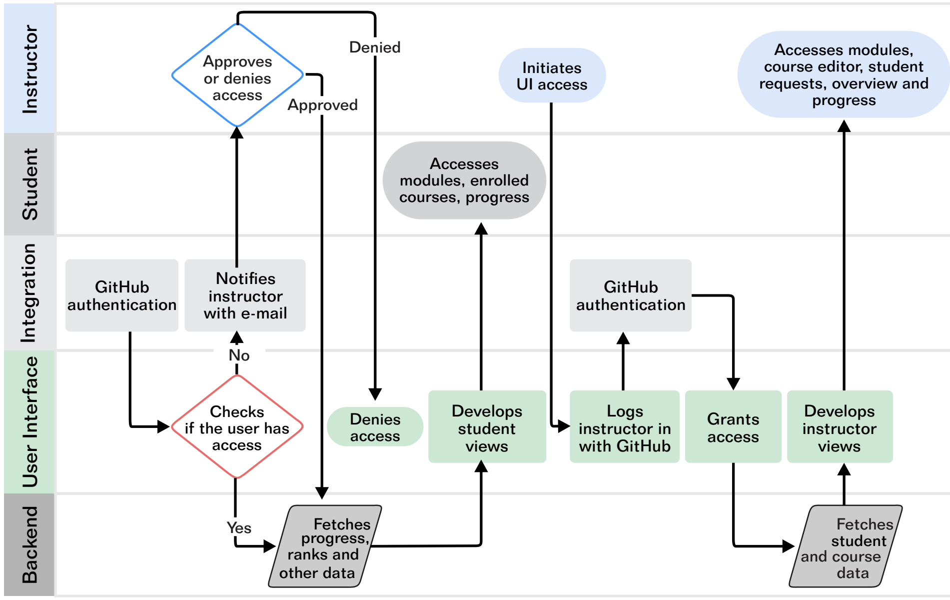


Fig. 5: MCD platform sequence flow for Student and Instructor user roles.

students' comprehension of the concepts, tools, and techniques covered at both the fundamental and advanced levels. Additionally, the assessment evaluates students' familiarity with the various scripts employed within the lab exercises. Subsequently, our auto-grader system grades the submitted answers and allocates medals and peer rankings to students, as depicted in Fig. 3. Each medal signifies the student's updated knowledge level. These knowledge levels are classified as follows:

- Bronze (Level 1): We merged the initial two categories of the knowledge dimension, factual and conceptual knowledge, to create the first tier, known as the Bronze level. At this level, students possess understanding of terminologies, classifications, categories, and some principles and theories relevant to Cloud DevOps concepts.
- Silver (Level 2): This level corresponds to the procedural category of the knowledge dimension. Students at this stage will demonstrate proficiency in subject-specific skills, algorithms, and a certain level of techniques. They will also understand when to apply appropriate procedures in the context of application development.
- Gold (Level 3): This level pertains to the metacognitive category, specifically strategic knowledge. At this level, students will exhibit mastery of the concepts and skills necessary for constructing applications utilizing Cloud DevOps principles.

4) Digitally Verifiable Credentials: We have integrated the MCD platform with the LinkedIn API to grant learners (virtual) certificates/badges that recognize their achievements and help them showcase their skills to their professional profiles. This MCD platform component is designed to motivate students by granting them certificates (digitally verifiable credentials) for individual courses or for completing the entire MCD learning modules catalog. The learners will be able to directly share the credentials to LinkedIn from the MCD platform in two ways: (i) open credential without verification ID, and (ii) credential with verification ID.

IV. Cloud DevOps Learning Modules

The learning modules within the MCD platform are crafted to provide a foundational understanding of prominent Cloud DevOps tools and technologies, including Kubernetes, KubeEdge, Jenkins, and SDNs. Students engage with related concepts through real-world application scenarios focused on automation and management, offering practical insights into problem-solving approaches. These modules draw heavily from previous research findings in fields such as video processing [45], cyber defense using pretense [46], cloud federation for trusted computing [47], and data pipeline automation for knowledge discovery [48], and machine learning for smart healthcare appointment scheduling [49]. Fig. 6 shows the mapping of various research-inspired application workflows to essential Cloud DevOps practices and tools on public and research cloud infrastructure resources. The documentation for the lab exercises can be freely accessed online at [50].

A. *Learning Modules Overview*

The MCD platform's learning modules are structured into eight sets of modules, accompanied by self-study activities aimed at facilitating effective comprehension of Cloud DevOps concepts across diverse cloud infrastructures. As students progress through these lab exercise manuals, they begin by acquainting themselves with relevant materials to grasp the fundamental concepts of specific technologies. Subsequently, students engage in hands-on experiences with the designated tools and technologies following guided instruction manuals. Upon completing the lab exercise steps, students undertake a post-lab technical assessment to assess their acquired skills and knowledge. Following the assessment, students receive medals, peer rankings, and gain insight into their standing relative to their peers.

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

Each lab exercise within the MCD platform showcases both new and existing functionalities across open/public cloud platforms, offering students comprehensive insights into their intricate integration, which proves valuable across a wide array of application scenarios. Every learning module on the MCD platform delves into a distinct focus pertaining to a specific technology, platform, and application use case (App). The available learning modules encompass: 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; Module-4) CI/CD with Jenkins with Data-intensive Healthcare Web App; Module-5) Tanzu Community Edition with Drone Video Streaming and Analytics App; Module-6) Ansible for Infrastructure as Code with Cybersecurity Operations in a Supply Chain/Logistics App; Module-7) Kubeflow on AWS with Health Informatics App; and Module-8) Prometheus and Grafana with Video Tranmission App. We discuss the last four modules in detail in this paper, while the first four modules can be found in our previous work [51].

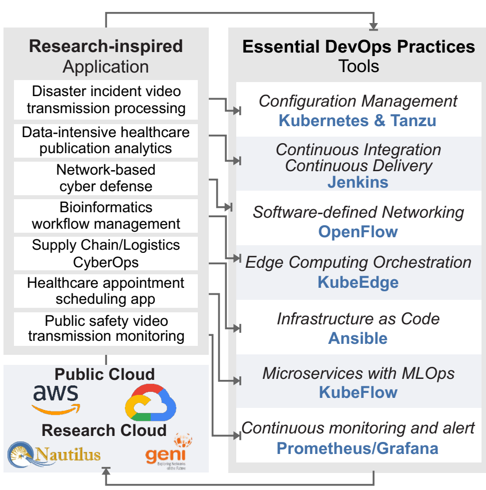


Fig. 6: Mapping of research-inspired applications to various DevOps practices and tools performed in public and research cloud platforms.

B. Module-5—VMWare Tanzu for Kubernetes Cluster Management

1) Technology: This module focuses on VMware Tanzu Community Edition (TCE), using Kubernetes platform for working in small-scale or pre-production environments. TCE enables creation of application platforms by leveraging a cluster API to provide declarative deployment and management of Kubernetes clusters. TCE also allows learners to get bootstrapped by providing a set of opinionated building blocks. Additionally, it enables them to add or replace these with their own components.

2) Platform: In this module, we use the AWS cloud infrastructure where Ubuntu Linux is spawned in an EC2 instance and Tanzu managed clusters are deployed with two components: a management cluster and a workload cluster.

3) Application Use Case: The application used in this module is an image analytics application that performs object detection/classification from traffic video streams. This application includes two components, a client and a server. The client streams video and sends video frames to the server, whereas the Server takes the video frames and performs image analytics to detect and classify the moving objects. These two components are implemented using microservices encapsulated in Docker containers, where they communicate via RESTful APIs using the python Flask [52] package. The students will deploy this image processing application in two pods in the TCE cluster, perform example video streaming and objection classification, and collect analytics results in the form of category/number of objects. The students will use deployment to create pods, and service to expose RESTful APIs so that the client can communicate with the server. Docker container images of both the client and server have been prepared, and students can use them directly in the cluster deployment.

C. Module-6—Ansible for Infrastructure as Code with CyberOps in a Supply Chain/Logistics App

1) Technology: This module focuses on Ansible as an automation tool for managing servers in the cloud. The students will learn how to use the AWS cloud platform to bring up a simple Ansible topology that has 3 nodes - 1 Control node and 2 Managed nodes and learn how Ansible can be used to automate processes, e.g., user management and software/library management. The students also create a LAMP stack that runs a web server on the Managed nodes.

2) Platform: Students will create and launch the Ansible engine and nodes using AWS EC2 services, and then configure the nodes to speak to each other using Secure Shell (SSH) concepts.

3) Application Use Case: The application use case for this module is a cybersecurity breach scenario in a supply chain/logistics application [53], where a node is made vulnerable by executing a script that creates different user accounts and then creates a group named *attack* and makes the new users part of it. The script will then assign all the permissions in the system to this group enabling the users to execute super user commands. This allows collection of internal data into a file which is then moved to a different node. In this use case, the students will then perform an audit of the system configuration. The baseline information on the system configuration is gathered from the first node through the Ansible Audit Playbook execution.

This whole set up works with a 3-node infrastructure, a master node with Ansible Engine and 2 worker nodes. One of the worker node works as an attacker and the other worker node works as the victim, both nodes will be administrated by Ansible. Ansible is used to determine any suspicious action in the nodes through an audit.

D. Module-7—Kubeflow with Health Informatics App

1) Technology: The module focuses on Kubeflow which helps students learn about machine learning (ML) operation

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

automation in the cloud using the Kubeflow framework. Kubeflow helps building, deploying, and managing multi-step ML workflows and ascribes to make it simple, portable and scalable. The complex systems are managed using Kubernetes and deployed to various clouds, local or on-premises platforms for experimentation for production usage. There are several components within Kubeflow such as Jupyter notebooks, ML training, hyperparameter tuning and serving workloads across multiple platforms.

2) Platform: Students will use the AWS cloud platform to perform this lab where they will use MiniKF, a third party service within the AWS platform to launch an instance to perform the ML tasks. In this module, students will work on Kubeflow that concentrates on the ML workflows accessed through MiniKF, which are used on top of Docker containers and uses Kubernetes to orchestrate them.

3) Application Use Case: The application use case for this module is an Health Informatics App that predicts the consultation length by accounting for its semi-continuous nature (i.e., zero in case of no-shows and positive otherwise, using ML algorithms), identifying important features for predicting no-shows, non-zero consultation lengths and assess the impact of integrating the ML-based prediction with the appointment scheduling system. The data used for this use case will be passed through a multi-step pipeline for various ML algorithms viz., Random Forest, Decision Trees. The model is designed such that the patients' data on their appointment set up and information on whether or not they have been coming for visits as per their schedule is utilized to make decisions on scheduling the future appointments. This also factors in double-booking of appointments and deferral of the patients based on the output.

E. Module-8—Prometheus and Grafana for Video Transmission App

1) Technology: This module focuses on Prometheus and Grafana monitoring tool stacks and their usability to monitor servers and applications. Prometheus is an open-source monitoring system with a dimensional data model, flexible query language, efficient time series database and alerting approaches. Grafana is a multi-platform open source analytics and interactive visualization web application.

2) Platform: The module deploys Prometheus and Grafana stack on the AWS cloud platform, familiarizes students with the basic concepts and functionalities using example Exporters to collect/visualize data from a sample application or server.

3) Application Use Case: The use case application for this module is based on a Video Transmission App. The students learn to develop an applicable Prometheus Exporter in Python, exporter will analyze the log file generated by video streaming server and exports the payload size, stream duration, throughput metrics for Prometheus and Grafana to scrape and visualize the data. The metrics to collect the data are configurable and temporal (i.e., in seconds, minutes, etc.) which are instantly reflected in Prometheus. Grafana is then connected to the Prometheus server to visualize the analytics.

V. 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 learner outcomes. In this section, we summarize the results obtained from participants with diverse backgrounds in terms of age, technical knowledge and work experience across the workshops. In addition, we detail results from a usability study with students with non-computer science background, and a series of focus groups with MCD platform users.

A. Knowledge Growth Study

A knowledge growth study was performed to evaluate the students' progress in exposure to important Cloud DevOps technologies and application use cases. Kubernetes is one of the early learning modules hosted in the MCD platform, and Kubeflow is one of the latest learning modules of the platform. The Knowledge growth studies for these two learning modules are discussed in the following:

1) Module-7—Kubeflow: In order to evaluate knowledge growth of participants, we performed a pre-lab evaluation survey and a post-lab technical assessment for 97 learners who successfully completed the Module-7. We performed a study similar to Module-1. Some example pre-lab questions for the first lab include:

- Are you familiar with Kubeflow?
- Are you familiar with machine learning?
- Have you used Amazon Web Services before?

The post-lab technical assessment comprises of multiple choice questions prepared based on the concepts and techniques used in Module-7 such as:

- What types of Machine Learning workloads can be handled by Kubeflow?
- What is the function of kfp.Client.create_experiment?
- The problem of predicting patient no-show risk will fall under which one of the following machine learning approaches?
- Given the class imbalance in the outcome variable (30% no-show, 70% show), which of the following metrics is not best suited for performance evaluation?

Based on the pre-lab assessment responses, participants were assigned knowledge levels as part of a statistical comparative study as seen in pre-lab evaluation column in Table II. Post-lab evaluation column shows the percentage of students in that level after completing the lab. The results show that 57% 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 at Level 2 or below could enhance their learning and move to Level 3 as we see a rise from 0% to 34%. This strongly demonstrates that the learning module completion certainly helped the participants to improve their skill level in the technologies explored in the Module-7.

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

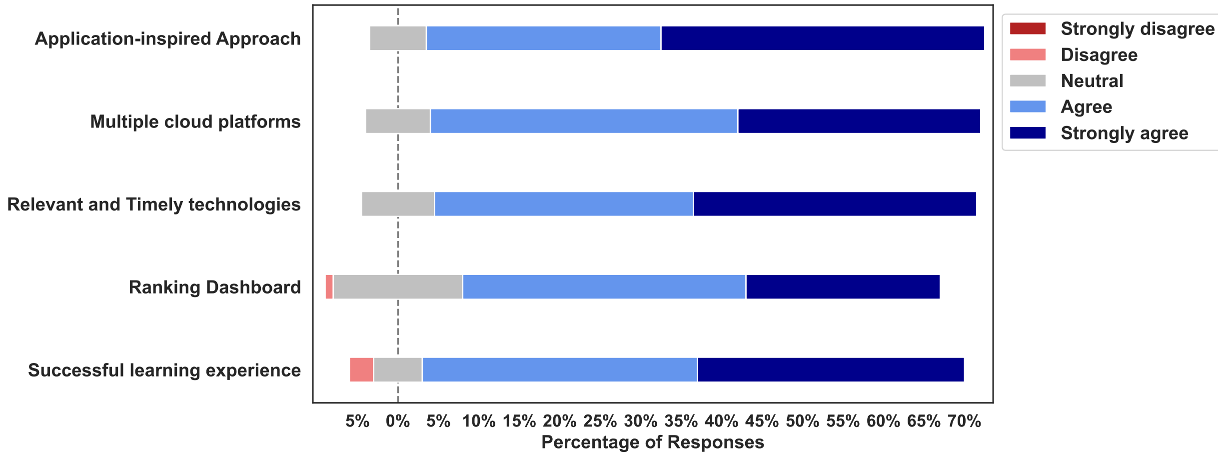


Fig. 7: Percentages of workshop attendees distributed with respect to their feedback on the provided questionnaire
.

TABLE II The distribution of students across different levels of knowledge before and after the successful completion of Module-7, as determined by pre-lab and post-lab assessments.

Knowledge Level	Pre-lab Evaluation	Post-lab Evaluation
0	37%	15.5%
1	48.5%	21.5%
2	14.5%	29%
3	0%	34%

B. Feedback on Workshop Outcomes

In the community workshops that were open to the public, we performed a feedback survey involving 76 randomly chosen participants on their opinion and improvement suggestions for the MCD platform. In the following, we show a sample set of 5 questions that were part of 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. 7. The Likert scale responses were mapped using 1-5 point scales (Strongly disagree - Strongly agree) for statistical analysis. We applied Spearman's Rank [54] and Kendall's tau [55] correlation analysis for the ordinal data format of the survey responses. Given that these factors were assessed using ordinal measurements, we use non-parametric statistics to examine their interrelations. The "Successful Learning Experience" is particularly used as an outcome variable, with other factors displaying moderately positive relationships, typically around the 0.5 mark. A value of 0.5 indicates a fair degree of association, falling between no relationship (0) and a perfect positive relationship (1). Although these variables play a role in fostering a successful learning experience, the correlation is not overly robust. We next employ the Chi-Square [56] test of independence to compare these factors against the outcome variable. Our analysis revealed the statistical significance, underscoring the contribution of these surveyed variables to a successful learning experience outcome.

From the findings presented, we ascertain that our strategy, which involves integrating real-world applications into the learning process of Cloud DevOps technologies, was well-received by the participants. They appeared enthusiastic about exploring the diverse public and research cloud platforms available through the MCD portal. Additionally, a majority of the attendees concurred that the technologies and principles covered in our learning modules are timely and relevant to cloud application development.

C. Usability Study with Students from Non-Computer Science Background

The usability study was performed by 4 independent evaluators from the Information Experience Lab at the University of Missouri, and involved students who were from industrial engineering and business backgrounds. The study involved participant observations and survey methods and included 8 (n = 8) randomly chosen participants. In the usability study, participants were asked to use Module-7 in the MCD platform (Kubeflow on AWS with Smart Scheduling App).

Table IV lists the participants' demographic information, which shows that all of the participants were from non-computer science backgrounds. Participants for this evaluation were purposefully sampled from a Data Engineering and Predictive Modeling class at the University of Missouri, consisting of undergraduate and graduate-level students in the Industrial Engineering program. After informed consent was obtained, the participants took part in a 75-minute session to complete a structured lab exercise within the module. Participants' interactions with the MCD platform were video recorded via

TABLE III Descriptive Statistics for the responses made in the Workshop Outcomes.

Survey Questions	Mean	Standard Deviation	Mode	$\tilde{\chi}^2$ p-value
1. Application-inspired Approach	4.44	0.67	5	6.78e-05
2. Multiple Cloud Platforms	4.27	0.66	4	0.029
3. Relevant and Timely	4.34	0.70	5	0.039
4. Ranking Dashboard	4.05	0.75	4	1.18e-07
5. Successful Learning Experience	4.15	0.88	5	-

TABLE IV Demographics of the Usability Study Participant (USP) pool with non-computer science background.

Participant	Age	Gender	Student Type	Department	Ethnicity
USP-1	21	Male	Full-time	Industrial engineering	White or Caucasian
USP-2	22	Male	Full-time	Industrial engineering	White or Caucasian
USP-3	22	Female	Full-time	Industrial engineering	White or Caucasian
USP-4	21	Female	Full-time	Industrial engineering	White or Caucasian
USP-5	22	Male	Full-time	Industrial engineering	Hispanic
USP-6	22	Female	Full-time	Industrial engineering	White or Caucasian
USP-7	23	Male	Full-time	Industrial engineering	White or Caucasian
USP-8	21	Female	Full-time	Industrial engineering	White or Caucasian

Open Broadcaster Software (OBS), an open-source software for screen recording and live streaming. During the evaluation, trained researchers took field notes of participant interactions within the MCD platform. The video recordings and the field notes were stored for later analysis. After participants finished the lab component of the study, they then filled the System Usability Scale (SUS) [57] and rated the overall user-friendliness of the MCD platform on a single item adjectival ease-of-use scale.

Using a multi-methods approach, qualitative data were analyzed using inductive and deductive methods, and quantitative data were analyzed to provide descriptive statistics. For qualitative analysis, the deductive analysis was performed using the coding scheme outlined by Van den Haak et al. [58] which categorizes usability issues along the lines of comprehensiveness, feedback, data entry, terminology, and layout. Inductive analysis was also conducted to identify themes related to learner experience in video recordings and field notes [59]. Identified usability issues and the emergent themes were refined into four overarching themes of instructions in the form of text and images, system-generated feedback, and technical terms. For quantitative analysis, SUS scores were determined based on the methods outlined by the author in [60]. An additional single-item user-friendliness scale was used to interpret the SUS results [57].

Quantitative results indicated an overall SUS score as 61.6 on a scale of 100 (where 100 is the highest). It means that the participants who were from non-computer science backgrounds rated the usability of the MCD platform as "D/Marginal" on a scale of A to F, where A is the best [61]. The "A-F" rating is a letter guide that explains the overall usability of the system. According to the SUS scores, participants rated the usability of the MCD platform as 63.7 and its learnability as 53.1. This suggests that the participants who did not have adequate pre-requisite computer science education found it difficult to learn the concepts, and note their experience of the MCD as an average learning system. Fig. 8 shows

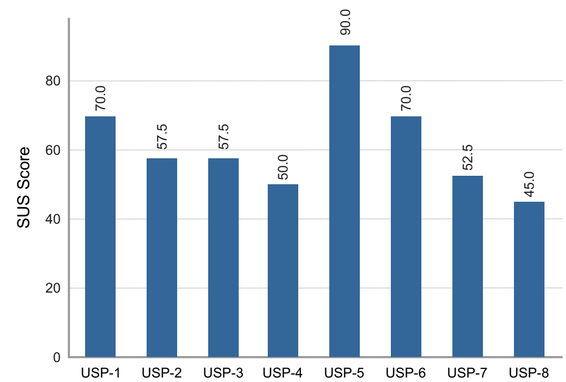


Fig. 8: Individual SUS scores of the usability test for Usability Study Participants (USP).

individual scores of each participant of the study. These ratings are slightly at odds with participants' ratings of the platform's user-friendliness, which was rated as "good" on a single-item adjectival ease-of-use scale [57]. These findings suggest that while participants found the MCD platform user-friendly, their lack of pre-requisite computer science background made the use of the MCD platform difficult to use or learn as they expected.

Meanwhile, qualitative findings from the usage test suggest that the participants liked the overall layout of tabs and chapters and the step-by-step format of instructions of the MCD platform. Across the sessions, no participant expressed dissatisfaction with the usage of the MCD platform or expressed a need to leave the study. All participants found the platform useful for their study curriculum and expressed a desire to use it in the future which suggests good acceptability. Overall, the participants' experience was satisfactory, although a few usability issues were encountered. The majority of the usability problems were related to the layout and design (font, color, images, etc.) of the instructions within the chapters, thoroughness of instructions in terms of step-by-step guidance of working with third-party applications (e.g., AWS), and

TABLE V Participants likes and dislikes identified during usability test for Usability Study Participants (USP).

Likes	**Participants**
Figures and images	USP-1, USP-3, USP-4, USP-5, USP-6
Step-by-step instructions	USP-1, USP-4, USP-5, USP-8
Video tutorial	USP-5, USP-7, USP-8
Instructions on the code	USP-4, USP-5
Overall layout of tabs and chapters	USP-3, USP-4, USP-8
The green and blue circles show the visibility of the student's progress through the chapters	USP-3, USP-4, USP-8
Dislikes	**Participants**
The instructions on the top of the instructional image are not clearly visible	USP-1, USP-7, USP-8
Screenshots appear to merge with the background because of the same color and lack of border/frame	USP-3, USP-4, USP-7
The instructions (annotations) within the images were of the same font size and color as the content on the images	USP-3, USP-4, USP-7
It is difficult to read the instructions as the layout of headings, figures, and text is not clear	USP-2, USP-3, USP-4, USP-5, USP-7
The screenshots are not updated with the current issues and errors	USP-3, USP-7
The lack of flexibility in moving back and forth between the chapters	USP-2, USP-3, USP-4, USP-5, USP-8
Worried about losing the current execution of the task if go back and forth between chapters while waiting for the environment to set up	USP-3, USP-7

notification for error prevention and regulation. For instance, the texts between headings and images needed to be more visual for learners. In this regard, USP-1 stated: "Some of these smaller texts are very easy to miss. I always read the big bold text."

Table V summarizes participants' likes and dislikes. Given that the participants were from non-computer science background, the participants also experienced difficulty understanding the technical terminology. USP-3 commented: "As someone who comes in with no background knowledge, maybe defining 'Docker', 'container' and the 'Kubernetes pods'." Further, it was observed that participants' interaction with third party applications (e.g., AWS Marketplace) for the purpose of completing lab exercise impacted their user experience with the MCD platform to a large extent. For instance, USP-1 stated, "I keep switching to different platforms. Maybe there is a way to all do that in the same website or like have that be set up for you or like have that be in a chapter."

D. Focus Groups Study

For the focus groups study, 6 users (n = 6) of the MCD platform were randomly selected to participate in the study. We divided them into two learner groups: Focus Group 1 included four participants who completed at least one course and Focus Group 2 included two participants who did not complete any course. In Focus Group 1, all participants (n=4) were computer science graduate students. For the Focus Group 2, we conducted a semi-structured interview with FGP-5 and FGP-6. See Table VI for participants' demographic information. Both qualitative and quantitative data were collected using multiple methods, and the results of the study are described below.

Participants' responses were analyzed using inductive methods. Results from inductive analysis suggested themes related to user experience of the MCD platform. Results suggest that participants found the MCD platform user-friendly and indicated that they were excited to learn and gain hands-on experience with Kubernetes and third party applications, e.g., AWS. Participants in both focus groups were asked to provide general comments on their reason to choose the MCD platform over other platforms available in the market. According to the participants from both sessions, the MCD platform was more suitable for students' needs. The self-paced format and short length of the modules was a strength. For instance, FGP-3 commented that "the courses are shorter, and up to the point, The explanation [of Kubernetes] is up to the point." The shortness and relevance of the courses was cited as being a motivational factor for participants to engage and complete the modules. While the participants in the second focus group had not completed any of the courses, they both provided insight into why. For example, FGP-5 was simply exploring the platform as a tool that he could make available to students in his own classes. FGP-6, who also had not completed any of the courses, indicated that shorter modules would motivate him to complete more of the MCD platform. When asked why he had not completed any of the courses he said that he was a full-time worker and student. Due to this he was looking for modules that were, "shorter and more digestible." In FGP-5's review of available platforms, he indicated that MCD was highly relevant and was among the best options that he had found on the market. FGP-2 indicated the relevance of MCD for students as it was free of cost in comparison to other platforms available in the market. For example, FGP-2 commented, "AWS has some courses but sometimes you have to pay for those." In addition, participants found the flow of the chapters and instructions in the course adequate. For example, FGP-5 commented specifically on the availability of lists of the courses with short descriptions on the homepage "as they build you up with the content."

Participants agreed that the assessments helped them to learn the concept of Kubernetes. For example, FGP-1 specifically commented that the practical examples and challenges in the assessment helped him to learn "how to use the Kubernetes tool through a practical application," although he felt that more

TABLE VI Demographics for the Focus Groups Participant (FGP) pool with various professions.

Focus Group #	Participant	Age (years)	Gender	Profession	Ethnicity
1 (Group)	FGP-1	24	Male	Software development engineer	White or Caucasian
1 (Group)	FGP-2	25	Male	Full-time student in Computer Science	White or Caucasian
1 (Group)	FGP-3	23	Male	Full-time student in Computer Science	Asian or Asian American
1 (Group)	FGP-4	23	Female	Full-time student in Computer Science	Asian or Asian American
2 (Group)	FGP-5	58	Male	Faculty	White or Caucasian
2 (Group)	FGP-6	27	Male	Software Developer and Aerospace Engineering graduate student	White or Caucasian

questions were needed to get a good average score. FGP-3 also noted that the MCD platform "mainly concentrated on the learning part" and the assessment had only a "few number of questions." Participants found the design of feedback received on the assessments was helpful to understand their learning progress. For example, FGP-4 indicated that the use of gold, silver, and bronze colors as knowledge level badges directed the attention of participants to where they stand in comparison to their peers. This was also highlighted by both FGP-5 and FGP-6 in Focus Group 2 with FGP-6 saying. "I liked that they kind of broke down the knowledge level... having gold, silver and bronze levels can be easily understood, and they provided information of what makes it a gold knowledge level versus a bronze. So, yeah, I think it was nice".

Participants found the MCD platform "modern, responsive, and very intuitive to use" (FGP-1). FGP-2 indicated that the platform had a user-friendly interface, however it was "a struggle going back [to the previous chapters within a module]. The page has only a forward button and doesn't have any back button." Overall, some participants felt that there were not enough questions in the assessment, but they appreciated the practical examples and challenges included in the assessment. Participants also found the feedback received on the assessments to be helpful in understanding their learning progress, and appreciated the use of colors to indicate knowledge level badges. These findings suggest that the assessments and feedback provided on the platform are effective in helping users learn about Kubernetes, but that improvements to the navigation and user interface of the platform could be beneficial.

Participants provided a number of recommendations for improvement and implementation of the platform. In line with the design goal, FGP-5 and FGP-6 noted the relevance of using GitHub as a sign-in platform for cloud computing and recommended continued use of GitHub for quick access to the platform. However, in Focus Group 1, FGP-1, FGP-2 and FGP-3 expressed concerns with the sign-in option using GitHub only. FGP-2 suggested using Google as an additional sign-on because non-computer science users "might have difficulties [signing-in into the MCD platform] because they don't usually have Github." FGP-1 and FGP-3 noted that the requirement of instructor's authentication to get access to the courses was tedious. In this regard, FGP-1 suggested using the university's portal, e.g., Canvas as an additional sign-in option for students.

In the second focus group, FGP-6 commented that the interface sometimes had issues when adapting to different browser resolutions. He said, "images didn't adjust their size well... the text would be overlapping with the images next to the course titles." FGP-6 also indicated that they wished that links to outside resources would open in a new tab.

VI. Study Implications and Limitations

A. Implications and applications

The findings of this study have profound implications for researchers and educators involved in the design and assessment of Cloud DevOps learning technologies. By combining qualitative user-driven evaluations with quantitative assessments based on established usability heuristics, a comprehensive understanding of user experiences is achieved, guiding the refinement of systems such as our MCD Platform. Qualitative feedback illuminates user preferences regarding content, functionality, and design elements, urging researchers to employ standardized qualitative research methodologies for deeper insights. Collaborative engagement between end-users and researchers is crucial for aligning learning technologies with usability standards and design principles, ensuring meaningful educational experiences for diverse users. Moreover, the study underscores the importance of comprehensible instructions in facilitating engagement, especially among non-computer science students. It highlights the potential of technological advancements to enhance educational experiences for various user groups while emphasizing the necessity of user-centric design principles. Moving forward, ongoing usability studies are essential for optimizing user experiences and sustaining the effectiveness of Cloud DevOps learning technologies in higher education. Through prioritizing user needs and human-centered design approaches, educators and researchers can collectively contribute to the creation of inclusive and impactful learning environments.

B. Limitations and opportunities

Acknowledging its limitations, this study utilized purposeful sampling from a single class within the Industrial Engineering program at the University of Missouri for the usability study, potentially limiting the generalizability of findings. To counterbalance this, efforts were made to diversify participant representation through randomly sampled focus groups, aiming to capture a wider range of perspectives on the MCD platform's usability. Additionally, the study employed multiple data collection methods, including observation, interviews, and surveys, to produce comprehensive and robust results on participants' experiences. While these measures address the limitation of sample homogeneity to some extent, future studies could benefit from further diversification of participant demographics to enhance the broader applicability of findings across various educational contexts and disciplines. Incorporating a more diverse participant pool would improve the external validity of research findings, providing insights that are more universally relevant and impactful for diverse learners.

VII. Conclusion

In this paper, we presented the Mizzou CloudDevOps (MCD) platform, an interactive, self-service oriented and user-friendly platform for online learning of Cloud DevOps concepts. The MCD platform features learning to use of state-of-the-art tools/technologies such as e.g., Kubernetes, KubeEdge, SDN using VXLAN, CI/CD with Jenkins, VMWare Tanzu (CE), Ansible, Kubeflow and Prometheus/Grafana in the context of real-world applications. The MCD platform comprises eight learning modules, with each module containing lab exercises demonstrating exemplary application use cases in container cluster orchestration, cyber defense, trusted edge computing, and data pipeline automation, drone video streaming and analytics, cybersecurity operations for security auditing, health informatics machine learning pipelines, and system log analysis for video transmission. The target audience for MCD include students with computer science as well as non-computer science backgrounds (e.g., industrial engineering, healthcare).

Evaluation results showed that students were able to significantly improve their knowledge relating to Cloud DevOps technologies as evaluated by the knowledge growth study. Furthermore, the Likert Scale-based feedback received from the learners inferred statistical significance of the different survey questions towards successful learning experiences amongst learners. Our usability study and focused groups study showed that students found the layout and user experience of the platform to be user-friendly. Further, learners found the self-paced format and feedback mechanisms to be helpful for improving their Cloud DevOps knowledge and skills.

As part of future work, efforts can be focused on integrating various Cloud DevOps concepts for showing synergies between multiple DevOps technologies together to achieve diverse application workflow expectations (i.e., ensuring that the highest quality of software and infrastructure (with minimal defects) are routinely created and maintained for data/compute-intensive applications). Future studies with participatory design approaches and a larger, more diverse sample will improve the generalizability of the usability and instructional design findings from the current work and ensure benefits to more students and instructors broadly involved in using Cloud DevOps for scientific workflows.

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] "DevOps stats and facts – all the numbers you might ever need on DevOps in 2022 [online]," Accessed: February 10, 2023. Available at https://kruschecompany.com/devops-statistics-and-facts/.

[3] "DevOps Market Size, Share, Analysis Report By Component (Solutions, Services), By Deployment (On-premise, Cloud), By Enterprise Size (Large Enterprises, Small Medium Enterprises), By Application (IT, Telecom, BFSI, Retail), And Region Forecast, 2022 - 2030 [online]," Accessed: February 10, 2023. Available at https://www.acumenresearchandconsulting.com/devops-market.

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

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

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

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

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

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

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

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

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

[13] "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.

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

[15] "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/.

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

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

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

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

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

[21] 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.

[22] "Tanzu Community Edition [online]," Accessed: November 02, 2022. Available at https://tanzucommunityedition.io/.

[23] "Red Hat Ansible [online]," Accessed: November 02, 2022. Available at https://www.ansible.com/.

[24] "Kubeflow, the machine learning toolkit for Kubernetes [online]," Accessed: November 02, 2022. Available at https://www.kubeflow.org/.

[25] "Prometheus [online]," Accessed: November 02, 2022. Available at https://prometheus.io/.

[26] "Grafana: The open observability platform [online]," Accessed: November 02, 2022. Available at https://grafana.com/.

[27] R. Chatley and I. Procaccini, "Threading devops practices through a university software engineering programme," in *2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 2020, pp. 1–5.

[28] Y. Demchenko, Z. Zhao, J. Surbiryala, S. Koulouzis, Z. Shi, X. Liao, and J. Gordiyenko, "Teaching devops and cloud based software engineering in university curricula," in *2019 15th International Conference on eScience (eScience)*. IEEE, 2019, pp. 548–552.

[29] M. Radenković, S. Popović, and S. Mitrović, "Project based learning for devops project based learning for devops: School of computing experiences," in *E-business technologies conference proceedings*, vol. 2, no. 1, 2022, pp. 127–131.

[30] F. Tuo, Y. Bai, S. Long, Y. Luo, T. Wang, R. Wang, and G. Yin, "A new model of docker-based e-learning in hadoop," in *Proceedings of the*

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

2018 International Conference on Distance Education and Learning, 2018, pp. 22–31.

[31] X. Bai, D. Pei, M. Li, and S. Li, "The devops lab platform for managing diversified projects in educating agile software engineering," in *2018 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2018, pp. 1–5.

[32] G. Rong, S. Gu, H. Zhang, and D. Shao, "Devopsenvy: an education support system for devops," in *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, 2017, pp. 37–46.

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

[34] J. B. Baillieul, L. O. Hall, J. M. Moura, S. S. Hemami, G. Setti, G. Grenier, M. B. Forster, F. Zappulla, J. Keaton, D. McCormick *et al.*, "The first ieee workshop on the future of research curation and research reproducibility," 2017.

[35] F. Zhao, X. Niu, S.-L. Huang, and L. Zhang, "Reproducing scientific experiment with cloud devops," in *2020 IEEE World Congress on Services (SERVICES)*. IEEE, 2020, pp. 259–264.

[36] M. De Bayser, L. G. Azevedo, and R. Cerqueira, "Researchops: The case for devops in scientific applications," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 1398–1404.

[37] A. Qumer Gill, A. Loumish, I. Riyat, and S. Han, "Devops for information management systems," *VINE Journal of Information and Knowledge Management Systems*, vol. 48, no. 1, pp. 122–139, 2018.

[38] L. de Aguiar Monteiro, D. S. M. P. Monteiro, W. H. C. Almeida, A. C. de Lima, and I. S. Sette, "Methods of implementation, maturity models and definition of roles in devops frameworks: A systematic mapping," in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, 2020, pp. 1766–1773.

[39] M. Sánchez-Gordón and R. Colomo-Palacios, "A multivocal literature review on the use of devops for e-learning systems," in *Proceedings of the Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality*, 2018, pp. 883–888.

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

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

[42] "What is Markdown, and why should you use it? [Online]," Available at *https://www.ultraedit.com/company/blog/community/what-is-markdown-why-use-it.html*.

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

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

[45] 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.

[46] 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.

[47] 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.

[48] 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.

[49] S. Srinivas and H. Salah, "Consultation length and no-show prediction for improving appointment scheduling efficiency at a cardiology clinic: a data analytics approach," *International Journal of Medical Informatics*, vol. 145, p. 104290, 2021.

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

[51] S. Wang, R. Neupane, A. Pandey, X. Cheng, and P. Calyam, "Online learning platform for application-inspired cloud and devops curriculum," in *2021 IEEE 28th International Conference on High Performance Computing, Data and Analytics Workshop (HiPCW)*. IEEE, 2021, pp. 35–42.

[52] "Flask [online]," Accessed: November 02, 2022. Available at https://flask.palletsprojects.com/en/2.2.x/.

[53] D. Smith and S. Srinivas, "A simulation-based evaluation of warehouse check-in strategies for improving inbound logistics operations," *Simulation Modelling Practice and Theory*, vol. 94, pp. 303–320, 2019.

[54] J. H. Zar, "Spearman rank correlation," *Encyclopedia of biostatistics*, vol. 7, 2005.

[55] R. Newson, "Parameters behind "nonparametric" statistics: Kendall's tau, somers' d and median differences," *The Stata Journal*, vol. 2, no. 1, pp. 45–64, 2002.

[56] M. L. McHugh, "The chi-square test of independence," *Biochemia medica*, vol. 23, no. 2, pp. 143–149, 2013.

[57] A. Bangor, P. Kortum, and J. Miller, "Determining what individual sus scores mean: Adding an adjective rating scale," *Journal of usability studies*, vol. 4, no. 3, pp. 114–123, 2009.

[58] M. Van Den Haak, M. De Jong, and P. Jan Schellens, "Retrospective vs. concurrent think-aloud protocols: testing the usability of an online library catalogue," *Behaviour & information technology*, vol. 22, no. 5, pp. 339–351, 2003.

[59] J. W. Creswell and C. Poth, "Qualitative inquiry &research design: Choosing among five approoaches, ter," *Ahmad Lintang Lazuardi, Cet. I*, 2015.

[60] J. Broak, "Sus: a 'quick and dirty'usability scale, usability evaluation in industry," 1996.

[61] J. Sauro, *A practical guide to the system usability scale: Background, benchmarks & best practices*. Measuring Usability LLC, 2011.

Roshan Lal Neupane received his MS degree in Computer Science from University of Missouri-Columbia. He is currently a Cyberinfrastructure Engineer at Cyber Education, Research and Innovation (CERI) Center with College of Engineering in University of Missouri. He is also pursuing PhD in Computer Science at the University. His research interests include cybersecurity, Blockchain, cloud/edge computing, internet of things, artificial intelligence, and cloud and network security. He is a student member of IEEE.

Prasad Calyam received his MS and PhD degrees from the Department of Electrical and Computer Engineering at The Ohio State University in 2002 and 2007, respectively. He is currently the Greg L. Gilliom Professor of Cybersecurity in the Department of Electrical Engineering and Computer Science at University of Missouri-Columbia. Previously, he was a Research Director at the Ohio Supercomputer Center. His current research interests include distributed and cloud computing, computer networking, and cybersecurity. He is a Senior Member of IEEE.

Songjie Wang received the MS degree from the Department of Electrical Engineering and Computer Science at the University of Missouri-Columbia in 2018, and the Ph.D. degree in plant science at the same university in 2009. He is currently a Cyber Infrastructure Engineer with the College of Engineering, University of Missouri-Columbia. His research interests include cloud and edge computing, computer networking, Internet of Things, cybersecurity, and big data.

Kiran Neupane received his MS degree in Data Science and Analytics from the University of Missouri-Columbia. He is working as a data research consultant with the Department of Economics in the University. His research interests include Data Engineering, Artificial Intelligence and Machine Learning, Statistical Modeling and Visualization, Cloud Computing, and Cybersecurity.

This article has been accepted for publication in IEEE Transactions on Learning Technologies. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TLT.2024.3428842

Ashish Pandey received his B. tech degree in Mechanical Engineering from Indian Institute of Technology, Jodhpur, India. He is currently pursuing his PhD degree in the Department of Electrical Engineering and Computer Science at University of Missouri-Columbia. His current research interests include cloud computing, recommender systems, artificial intelligence, machine learning, and bioinformatics.

Xiyao Cheng received her MS degree in Computer Science from China Agriculture University. She is currently pursuing Ph.D. degree in the Department of Electrical Engineering and Computer Science at University of Missouri-Columbia. Her current research interests include knowledge graph based data modeling and conversational agents.

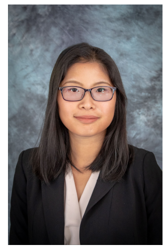
Durbek Gafurov received his Bachelor's degree from Politecnico di Torino. He is currently pursuing his MS degree in the Department of Electrical Engineering and Computer Science at the University of Missouri-Columbia. His current research interests include network and compute-intensive applications.

Hemanth Sai Yeddulapalli received his BTech degree from G. Pulla Reddy Engineering College, Kurnool. He is currently pursuing MS degree in Department of Electrical Engineering and Computer Science at University of Missouri-Columbia. His research interests include Cloud Computing and Machine Learning.

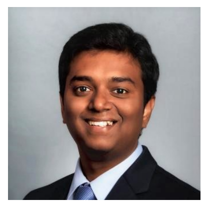
Noah Glaser received his MA and PhD degrees from the Department of Educational Studies at the University of Cincinnati in 2016 and 2020, respectively. He is currently an Assistant Professor at the University of Missouri-Columbia's School of Information Science & Learning Technologies, where he is also the director of the Information Experience Lab. His research interests include design research to create learning interventions that use innovative technologies such as virtual reality, video games, mobile devices, and artificial intelligence.

Kanu Priya Singh is a Postdoctoral Associate at the XRPediatrics Lab, housed in the Yale Center of Immersive Technologies in Pediatrics at Yale School of Medicine where she is working on the user-centered design and development of a virtual reality-based videogame for substance use prevention in adolescents. Kanu earned her Ph.D. in Information Science and Learning Technologies from the University of Missouri-Columbia in 2023. She received her MS degree in Information Systems from Drexel University in 2011. Before joining her Ph.D. studies in 2019, Kanu had worked as an Information Technology process consultant at Accenture Solutions in India. Her research interests broadly include user centered design and development of digital technologies and interactive experiences.

Yuanyuan Gu received her MA degree in TESOL (Teaching English to Speakers of Other Languages) from Stony Brook University in New York and MS degree in Learning Technologies and Design from University of Missouri-Columbia. She is currently pursuing Ph.D. degree in the School of Informational Science & Learning Technologies at University of Missouri-Columbia. Her research interests include game-based learning, virtual reality enabled learning environment and learning experience design.



Shangman (Eunice) Li received her Master of Education from Colorado State University. She is currently a Ph.D. candidate in the School of Information Science & Learning Technologies at the University of Missouri-Columbia. Her research interests include, user experience study, immersive learning, and accessibility. She is an active member of AECT (Association of Educational Communications and Technology) and AERA (American Educational Research Association).



Sharan Srinivas received his MEng and PhD degrees in Industrial Engineering and Operations Research from the Pennsylvania State University, USA. He is currently an Assistant Professor with a joint appointment in the Department of Industrial and Manufacturing Systems Engineering and the Department of Marketing at the University of Missouri. He is also a core faculty member at the MU Institute for Data Science and Informatics. His area of specialization is data analytics and operations research with research interests in healthcare operations management, transportation/logistics, smart service systems and supply chain.