

# CONTAINER HÓA - PHƯƠNG PHÁP TRIỂN KHAI NHANH HỆ THỐNG QUẢN LÝ HỌC TẬP (LMS) TRÊN MÔI TRƯỜNG ĐÁM MÂY

● PHẠM NGỌC DUY

## TÓM TẮT:

Trong thời đại Công nghiệp 4.0, nhằm hướng đến xu thế chuyển đổi số, các tổ chức doanh nghiệp hoạt động trong mọi lĩnh vực đều cần ứng dụng mạnh mẽ các giải pháp công nghệ thông tin (CNTT) tiên tiến để xây dựng và triển khai các hệ thống quản lý một cách nhanh chóng và hiệu quả trên môi trường phi truyền thống như đám mây. Không chỉ được sử dụng phổ biến trong môi trường giáo dục, hệ quản lý học tập (LMS) hiện đang hướng đến các tập đoàn, doanh nghiệp nhằm thực hiện chiến lược đào tạo và phát triển của tổ chức.

Nhằm đề xuất phương pháp triển khai nhanh chóng hệ thống LMS trên môi trường đám mây thông qua việc ứng dụng công nghệ container hóa, bài viết giới thiệu triển khai hệ thống LMS Moodle trên các môi trường bằng cách ứng dụng nền tảng container Docker. Đánh giá cho thấy, container giúp triển khai hệ thống LMS nói riêng cũng như các hệ thống phần mềm nói chung nhanh chóng và hiệu quả trên đa nền tảng, từ tại chỗ cho đến đám mây.

**Từ khóa:** Container hóa, Docker, điện toán đám mây, hệ thống quản lý học tập.

## 1. Đặt vấn đề

Trong thời đại kỹ nguyên số như hiện nay, các tổ chức hoạt động trong mọi lĩnh vực, từ kinh tế đến giáo dục, dần hướng tới mục tiêu tin học hóa, số hóa toàn hệ thống. Doanh nghiệp, trường học cần ứng dụng mạnh mẽ các giải pháp CNTT để xây dựng và triển khai những hệ thống hỗ trợ quản lý và vận hành tổ chức một cách nhanh chóng và hiệu quả.

Không chỉ được sử dụng phổ biến trong môi trường đào tạo bậc cao như đại học, hệ quản lý

học tập (LMS) hiện đang hướng đến các tập đoàn, doanh nghiệp nhằm thực hiện chiến lược đào tạo và phát triển của tổ chức. Vấn đề đặt ra là, cần có các giải pháp để những hệ thống phần mềm của tổ chức như LMS được triển khai một cách hiệu quả nhất, nhanh chóng và dễ dàng, nhưng vẫn đảm bảo việc vận hành hệ thống diễn ra thông suốt, tương thích với môi trường triển khai. Trong giai đoạn dịch COVID-19 đang diễn ra phức tạp tại Việt Nam và trên thế giới, nhu cầu triển khai các hệ thống học trực tuyến và LMS một cách nhanh

chồng tại các trường học, đơn vị cơ sở giáo dục lớn là rất lớn.

Công nghệ ảo hóa là một trong những công nghệ cốt lõi của điện toán đám mây (ĐTĐM), thực hiện trừu tượng hóa và phân chia các tài nguyên vật lý thành các tài nguyên ảo, còn các đám mây quản lý thì giúp phân bổ các tài nguyên điện toán ảo đó. Ảo hóa giúp môi trường điện toán độc lập với cơ sở hạ tầng vật lý, trong khi dịch vụ bởi ĐTĐM cung cấp tài nguyên điện toán dùng chung theo yêu cầu thông qua Internet.

Công nghệ container hóa ra đời sau ảo hóa, hiện đang là xu hướng công nghệ nổi bật, đặc biệt đối với việc xây dựng, triển khai và các hệ thống phần mềm ứng dụng [1,2]. Công nghệ container hóa thực hiện một cơ chế dạng ảo hóa nhưng ở cấp độ hệ điều hành, đóng gói các phần mềm, dịch vụ cùng với những gì cần thiết (như mã chương trình, thư viện, gói cài đặt,...) để ứng dụng có thể hoạt động. Tất cả gói gọn trong một "container" để các container có thể hoạt động ở bất kỳ môi trường nào, từ máy tính, máy chủ truyền thống cho đến đám mây.

Container nhỏ, nhanh, có tính di động, hoạt động độc lập và nhất quán trên các môi trường. Do đó, công nghệ container hóa cho phép phát triển và triển khai phần mềm ứng dụng một cách hiệu quả, nhanh chóng [1,3].

Bài báo đề xuất phương pháp ứng dụng container hóa trong triển khai hệ thống LMS một cách nhanh chóng và dễ dàng trên môi trường tại chỗ và đám mây, với việc triển khai một hệ thống LMS Moodle trên đám mây AWS sử dụng nền tảng container hóa Docker.

## 2. Hệ thống quản lý học tập (LMS)

Hệ thống quản lý học tập hay hệ quản trị đào tạo (tiếng Anh: Learning Management System, viết tắt LMS) là một phần mềm ứng dụng để quản trị, tài liệu hóa, theo dõi, báo cáo và phân phối các khóa học giáo dục, chương trình đào tạo hay các chương trình đào tạo và phát triển. Khái niệm hệ thống quản lý học tập xuất phát từ "học tập trực tuyến" (e-Learning). Ban đầu LMS xuất hiện trong lĩnh vực giáo dục bậc cao, nhưng ngày nay phần lớn các LMS lại đang tập trung vào thị trường doanh nghiệp. LMS chiếm phần khúc lớn nhất trong thị trường hệ thống học tập. Đối với thị trường giáo dục

bậc cao, 3 hệ thống LMS dẫn đầu về số lượng tổ chức sử dụng cũng như số học viên theo học là: Blackboard, Canvas và Moodle. Trong khi Blackboard, Canvas được ưa chuộng tại Hoa Kỳ (~30% thị trường với mỗi phần mềm) thì Moodle lại chiếm tỉ lệ lớn (trên 50%) ở thị trường khác.

Hệ thống quản lý học tập là sự kết hợp của hai từ - học tập và quản lý. "Học tập" đề cập đến việc cung cấp các khóa học và chương trình đào tạo, trong khi "quản lý" đề cập đến việc tổ chức các khóa học và nội dung này theo cách giúp người học dễ dàng sử dụng. Sự đa dạng các chức năng và ứng dụng của LMS phụ thuộc vào mục tiêu của tổ chức. Tuy nhiên, chức năng phổ biến nhất là phân phối và theo dõi các khóa học điện tử. Mỗi LMS giúp các giáo viên dạy tất cả những tài liệu đào tạo của họ lên một nơi để mọi người có thể truy cập chúng từ xa. Một số phần mềm LMS cũng có các công cụ tích hợp để phát triển tài liệu mà không cần sử dụng phần mềm của bên thứ ba.

Về cơ bản, các hệ thống LMS là kho lưu trữ lớn các khóa học và nội dung đào tạo sử dụng mà học viên có thể truy cập và theo dõi.

Về phía sử dụng: Có người dùng truy cập các khóa học và những người cung cấp, cập nhật nội dung đào tạo, thường chính là các giảng viên. Hệ thống quản lý học tập trực tuyến có thể được sử dụng bởi cả các tổ chức giáo dục và các công ty. Ví dụ, các doanh nghiệp có thể sử dụng các tài nguyên đào tạo trực tuyến cho các kênh bán hàng và dịch vụ khách hàng của họ.

Về triển khai, vận hành kỹ thuật: Phần lớn các LMS hiện đại đều trên nền web (web-based LMS). Ban đầu, các LMS được thiết kế để được lưu trữ cục bộ tại chỗ (on-premise) - tổ chức mua bản quyền cho một phiên bản phần mềm và cài đặt trên các máy chủ và mạng riêng của tổ chức. Hiện nay, với sự phát triển của ĐTĐM và công nghệ ảo hóa, nhiều LMS được triển khai và lưu trữ trên các máy chủ ảo hoặc trên đám mây (on-cloud) bởi các nhà cung cấp hạ tầng đám mây. Ngoài ra, nhiều LMS hiện đang được cung cấp dưới dạng SaaS (phần mềm-dưới-dạng-dịch-vụ) với dịch vụ lưu trữ hoặc toàn bộ phần mềm được cho thuê bởi các nhà cung cấp phần mềm LMS.

(1) LMS trên đám mây (Cloud-based LMS): Các nền tảng LMS này được lưu trữ (host) trên đám

máy. Tại đây, nhà cung cấp dịch vụ đám mây sẽ chịu trách nhiệm quản lý, bảo trì hệ thống, nâng cấp và cập nhật các công nghệ. Đối với các LMS dạng SaaS, tổ chức cũng như người dùng LMS không cần quan tâm đến các thiết bị phần cứng hay cài đặt bất kỳ phần mềm nào, họ có thể bắt đầu sử dụng hệ thống ngay chỉ cần có kết nối mạng Internet.

(2) LMS tại chỗ (on-premise/self-hosted LMS): Nhà cung cấp có thể hỗ trợ người dùng tải bộ cài đặt LMS từ một nguồn như website hoặc đĩa cài đặt. Với các nền tảng LMS tự lưu trữ, tổ chức hay người dùng có khả năng kiểm soát và tùy biến hệ thống hơn. Tuy nhiên, yêu cầu tổ chức phải có chuyên viên về CNTT để triển khai, vận hành hệ thống.

(3) LMS dạng ứng dụng trên máy tính/thiết bị di động: LMS có thể được cài đặt và truy cập từ nhiều thiết bị, bao gồm thiết bị di động. Điều này giúp toàn bộ nhóm e-learning dễ dàng hợp tác và theo dõi quá trình khi đang di chuyển.

Phần mềm LMS Moodle: Moodle là giải pháp quản lý học tập mã nguồn mở rất phổ biến để phân phối các khóa học và chương trình e-learning. Moodle có giao diện đơn giản: Các tính năng kéo và thả, các quyền dựa theo vai trò (role), báo cáo sẵn, hỗ trợ nhiều bản dịch ngôn ngữ, API được cung cấp cùng nhiều tài liệu hướng dẫn, v.v... Moodle sẵn sàng đáp ứng nhu cầu của bất kỳ tổ chức quy mô nào, với các tính năng chính bao gồm: (1) Xây dựng các khóa học (course) và thêm nội dung cho phần mềm; (2) Tạo các bài tập (assignment), bài kiểm tra (test) và câu hỏi (quiz) cho học viên; (3) Xem bài nộp (submission) của học viên và theo dõi tiến độ; (4) Thêm thời hạn khóa học vào lịch được chia sẻ; (5) Tạo các báo cáo về sự tham gia của học sinh; (6) Định nghĩa các quy trình công việc để chấm điểm bài tập và làm việc - với những người hướng dẫn/giáo viên.

### 3. Điện toán đám mây và công nghệ Container hóa

#### 3.1. Điện toán đám mây

Theo định nghĩa của NIST, "Điện toán đám mây là mô hình dịch vụ cho phép người dùng truy cập nhóm các tài nguyên điện toán dùng chung (ví dụ như mạng, máy chủ, lưu trữ, ứng dụng và dịch vụ) thông qua kết nối mạng một cách thuận tiện.

mọi lúc mọi nơi, theo yêu cầu; tài nguyên điện toán này cho phép người dùng tạo hoặc hủy nhanh chóng mà không cần sự can thiệp của nhà cung cấp dịch vụ". Điện toán đám mây là mô hình cho phép sử dụng tài nguyên điện toán - như máy chủ, quản lý cơ sở dữ liệu, lưu trữ dữ liệu, mạng, ứng dụng phần mềm - qua mạng Internet - theo yêu cầu, đối lập với việc sở hữu và tự vận hành các tài nguyên này tại chỗ (on-premise). So với mô hình CNTT truyền thống, ĐTĐM cung cấp cho các tổ chức nhiều lợi ích như: Hiệu quả chi phí (khi chỉ cần trả tiền những tài nguyên được sử dụng); tốc độ phát hành các ứng dụng, dịch vụ nhanh hơn; khả năng mở rộng/co giãn, giá cả phải chăng, triển khai trên quy mô toàn cầu.

ĐTĐM đối với giáo dục: Sự phát triển của ĐTĐM đã đem đến cho thị trường giáo dục trực tuyến "loại hình LMS dựa trên đám mây với nhiều ưu điểm như tiết kiệm chi phí, khả năng triển khai nhanh chóng, hệ thống được vận hành, cập nhật và bảo trì chuyên nghiệp, khả năng tùy chỉnh và mở rộng hệ thống linh hoạt...". Nguyễn Việt Dũng, Lê Huy Hoàng (2019) đánh giá, thay vì được cài đặt và vận hành trên hạ tầng CNTT nội bộ của các trường học như trước, LMS có thể được cài đặt và vận hành trên hạ tầng của nhà cung cấp dịch vụ đám mây với dịch vụ cơ sở hạ tầng IaaS hoặc sử dụng thông qua dịch vụ phần mềm dạng SaaS trọn gói.

#### 3.2. Công nghệ container hóa và Docker

Container là một gói có khả năng thực thi độc lập, nhẹ (lightweight), chứa đầy đủ (và chỉ vừa đủ) mọi thứ mà ứng dụng cần để hoạt động bao gồm mã chương trình (code, binary), các công cụ hệ thống, thư viện hệ thống (library), các gói phụ thuộc (dependency) - bất cứ thứ gì có thể cài đặt trên máy chủ [1,2]. Bằng cách đóng gói và cô lập mọi thứ trong một container, đảm bảo các container (ứng dụng được container hóa) sẽ luôn hoạt động giống nhau bất kể đang chạy trên môi trường nào. Container loại bỏ sự phụ thuộc giữa các ứng dụng và lớp cơ sở hạ tầng căn bản, cung cấp những khả năng mới về tính di động và tính hiệu quả, đặc biệt trong việc phát triển và triển khai phần mềm ứng dụng.

Ưu điểm chính của các container, đặc biệt là so với máy ảo, là cung cấp một cấp độ trừu tượng hóa, khiến chúng nhẹ và có thể di động. "Ảo hóa loại bỏ

sự cần thiết của toàn bộ một máy chủ cho một ứng dụng. Container hóa loại bỏ sự cần thiết của toàn bộ một hệ điều hành (OS) cho mỗi ứng dụng. Nhanh, khả năng di động, nhanh chóng và linh hoạt, có lập lỗi, dễ dàng quản lý và bảo mật là những lợi thế của việc sử dụng công nghệ container hóa so với mô hình máy chủ truyền thống cũng như công nghệ ảo hóa [2].

Docker là một nền tảng mã nguồn mở để xây dựng, triển khai và quản lý các ứng dụng được container hóa. Về cơ bản, đây là một bộ công cụ cho phép các nhà phát triển có thể xây dựng, triển khai, chạy, cập nhật và dừng các container bằng cách sử dụng các lệnh đơn giản cùng sự tự động hóa để tiết kiệm công sức làm việc. Docker được phát hành vào năm 2013 và đang là nền tảng container phổ biến nhất hiện nay [3]. Việc ứng dụng nền tảng container, theo Docker, mang lại giá trị lập tức cho tổ chức bằng cách giảm chi phí cơ sở hạ tầng và bảo trì để hỗ trợ các ứng dụng hiện tại, trong khi đẩy nhanh thời gian phát hành và triển khai các giải pháp mới.

Một số kết quả đem lại như: (1) Thời gian ra mắt thị trường nhanh hơn; (2) Năng suất phát triển ứng dụng tăng; (3) Tăng tốc triển khai; (4) Giảm bớt yêu cầu về cơ sở hạ tầng CNTT; (5) Tăng hiệu quả hoạt động CNTT; (6) Giải quyết các vấn đề nhanh hơn. Ngoài ra, việc sử dụng container giúp cải thiện chất lượng ứng dụng, giảm chi phí phát triển và thời gian gián đoạn trong ứng dụng, tạo điều kiện đổi mới trải nghiệm người dùng. Container hỗ trợ tốt cho các dịch vụ web, cơ sở dữ liệu; Hỗ trợ nền tảng đám mây và di trú dễ dàng [3].

#### 4. Triển khai hệ thống

Mô tả quá trình triển khai thực nghiệm: Tác giả tiến hành xây dựng container cho hệ thống LMS Moodle sử dụng nền tảng container hóa Docker và triển khai hệ thống LMS trên nhiều môi trường (tạ chỗ on-premises và đám mây on-cloud). Tiến hành triển khai phần mềm LMS trên một máy chủ truyền thống (on-premises) theo: (1) Cách thông thường (native); (2) thông qua ảo hóa (sử dụng KVM - Kernel-based Virtual Machine - một giải pháp ảo hóa đầy đủ cho Linux); và (3) thông qua container (sử dụng Docker). Từ đó, có những so sánh, đánh giá thực tế về container hóa với hệ thống truyền thống. Tác giả cũng tiến hành so sánh các thông số

về dung lượng chiếm dụng thiết bị lưu trữ cũng như thời gian triển khai và vận hành của (1), (2) và (3). Cuối cùng, triển khai (cài đặt và vận hành) hệ thống LMS Moodle cơ bản trên môi trường đám mây (AWS), sử dụng nền tảng container Docker.

#### 4.1. Triển khai hệ thống LMS Moodle trên máy chủ với container Docker

Hình 1: Hệ thống LMS Moodle trên máy chủ với container Docker

CPU	Intel Core i7-9700k @3.60Ghz
RAM	8 GSkil 32GB (4x8GB) DDR4-3000MHz
SSD	Samsung 970 EVO 512GB
Mainboard	Gigabyte Z390
Network	Gigabit Ethernet @1Gbps Internet connection bandwidth (U 100 /D: 100 Mbps)
OS	Ubuntu Server Bionic (18.04.4) LTS 64-bit

a. Chuẩn bị môi trường thử nghiệm: Cấu hình máy chủ được cho bằng thông số. Cài đặt máy thử nghiệm: (1) Cài đặt OS (Ubuntu Server 18.04.4 LTS) trên máy chủ; (2) Cập nhật hệ thống, cấu hình và cài đặt các gói hỗ trợ.

b. Triển khai hệ thống LMS Moodle trên máy chủ thực (bare-metal) với Docker container.

- Bước 1: Cài đặt Docker Engine (CE)

Tải và cài đặt gói docker-ce với lệnh: '\$ sudo apt-get install docker-ce'

- Bước 2: Chuẩn bị Docker image của Moodle:

Có thể sử dụng các image có sẵn được lưu trữ trên các kho hoặc tạo image mới với lệnh docker build. Tác giả sử dụng Docker image cho Moodle của Bitnami làm image cơ sở (Bitnami là một dạng hệ thống quản lý gói, cung cấp một thư viện các bộ cài đặt hoặc các gói phần mềm cho các ứng dụng web, các stack giải pháp cũng như các ứng dụng ảo). Image này được xây dựng trên image "minideb" - một image cơ sở Debian dạng tối thiểu. Có thể truy cập và kéo Docker image cho Moodle này từ trên kho Docker Hub hoặc thông qua GitHub. Việc tải image này (bitnami/moodle) về kho image trên hệ thống cục bộ một cách chủ động được thực hiện thông qua lệnh: '\$ docker pull bitnami/moodle'. Việc xây dựng image thủ công từ đầu hoặc thay đổi image có thể được thực hiện thông qua Dockerfile. Đối với Docker image cho Moodle, nội dung (tóm tắt) "Dockerfile" như Hình 2. Nếu cần tùy biến image hoặc tạo các image mở

rộng từ image ban đầu (trường hợp này sử dụng image "moodle"). Có thể thay đổi hoặc tạo mới Dockerfile. (Hình 2)

(Cách 2) Chạy ứng dụng container thì công: Tạo một network mới cho ứng dụng (Moodle) và cơ sở dữ liệu (MariaDB). Tạo một volume (dạng

Hình 2: Nội dung Dockerfile, docker-compose.yml và các lệnh docker-run

<pre>FROM docker.io/bitnami/mariadb:latest LABEL maintainer="Bitnami &lt;containers@bitnami.com&gt;" ENV PATH="/usr/sbin" COPY prebuilds/ / # Add doc and perl libs and go perl modules as they RUN install_packages ... RUN /build/bitnami-user.sh RUN /build/install-mariadb RUN bitnami-ops unpack apache-2 # 43.0 -checksum RUN bitnami-ops unpack php-7.3.16.0 -checksum ... ... RUN apt-get update &amp;&amp; apt-get upgrade -y &amp;&amp; \ ... COPY /moodle/ / # Khai báo và khai tạo các biến môi trường ENV APACHE_HTTP_PORT_NUMBER="80" \     BITNAMI_APP_NAME="moodle" \     MARIADB_HOST="mariadb" \     MOODLE_DATABASE_NAME="moodle_database" \     MOODLE_EMAIL="dvsyn@vmaru.edu.vn" \     MOODLE_SITE_NAME="NCKH2019 MoodleLMS" \     MOODLE_USER_NAME="nckh2019" \     ... DPOSE 80443 ENTRYPOINT [ "/app-entrypoint.sh" ] CMD [ "/run.sh" ]</pre>	<pre>version: '2' services:   # tạo 2 container sử dụng 2 image tương ứng   mariadb:     image: bitnami/mariadb:10.1 # image cơ sở dữ liệu     environment: # các biến môi trường cho mariadb       - MARIADB_USER=moodle_db_nckh2019       - MARIADB_DATABASE=moodle_db     volumes:       - mariadb_data:/bitnami   moodle:     build:       image: bitnami/moodle:3 # image ứng dụng Moodle     environment: # các biến môi trường cho moodle       - MARIADB_HOST=mariadb       - MOODLE_DATABASE_USER=moodle_db_nckh2019       - MOODLE_DATABASE_NAME=moodle_db     ports:       - "80:80"       - "443:443"     volumes:       - moodle_data:/bitnami     depends_on:       - mariadb   volumes:     # tạo 2 volume cho 2 container     mariadb_data</pre>	<pre>\$ docker network create moodle-net \$ docker volume create --name mariadb_data \$ docker run -d --name mariadb \   --ALLOW_EMPTY_PASSWORD=yes \   --MARIADB_USER=moodle_db_nckh2019 \   --MARIADB_DATABASE=moodle_db \   --net moodle-net \   --volume mariadb_data:/bitnami \   bitnami/mariadb:latest \$ docker volume create --name moodle_data \$ docker run -d --name moodle_container --E0:80 --443:443 \   --ALLOW_EMPTY_PASSWORD=yes \   --MOODLE_DATABASE_USER=moodle_db_nckh2019 \   --MOODLE_DATABASE_NAME=moodle_db \   --MOODLE_DATABASE_PASSWORD="" \   --MOODLE_EMAIL=dvsyn@vmaru.edu.vn \   --MOODLE_LANGUAGE=en \   --MOODLE_SITE_NAME=NCKH2019 MoodleLMS \   --MOODLE_SMP_INSTANCE="no" \   --MOODLE_USER_NAME="nckh2019" \   --net moodle-net \   --volume moodle_data:/bitnami \   bitnami/moodle:latest</pre>
"Dockerfile"	"docker-compose.yml"	Các lệnh docker-run chạy container

- Bước 3: Chạy Docker container từ các image.

Đối với hệ thống Moodle, cần chạy 2 dịch vụ chính tương ứng với 2 image là image cơ sở dữ liệu (sử dụng MariaDB) và image ứng dụng Moodle. Tái nhiên, có thể tạo duy nhất một image chung cho cả hệ thống gồm cơ sở dữ liệu và ứng dụng. Nhưng theo quan điểm phát triển phần mềm tiên tiến, việc chia nhỏ các dịch vụ trong hệ thống sẽ khiến cho quá trình phát triển, cập nhật nâng cấp ứng dụng hiệu quả hơn.

Khi thực hiện lệnh docker run, Docker sẽ tạo container từ (các) image. Docker sẽ kiểm tra xem đã có sẵn image trên hệ thống cục bộ hay chưa. Trong trường hợp chưa có, Docker sẽ tìm kiếm (pull) tự động image từ kho theo tên image đã được khai báo. Việc tạo và chạy Docker có thể được thực hiện thủ công với lệnh docker run kèm các tham số hoặc được thực hiện tự động thông qua Docker compose.

(Cách 1) Tạo và chạy container tự động với docker-compose: Cấu hình cài đặt được lưu trong file "docker-compose.yml". Tạo file với nội dung (tóm tắt) như Hình 2 hoặc tải file có sẵn. Sau đó thực hiện lệnh: '\$ docker-compose up -d'

persistence) và tạo container MariaDB. Tạo một volume (dạng persistence) và tạo container Moodle: Nội dung các câu lệnh trong Hình 2.

- Bước 4: Kiểm tra hoạt động của hệ thống và các container. Kiểm tra danh sách và trạng thái các container đang chạy, sử dụng lệnh: '\$ docker ps. Hiện đang có 2 container vừa tạo là "Mariadb" và "Moodle" đều đang hoạt động.

c. Thiết lập và vận hành hệ thống LMS Moodle: Để truy cập vào hệ thống Moodle vừa được triển khai, có thể sử dụng trình duyệt và truy cập tới địa chỉ IP hoặc tên miền của hệ thống (hệ thống Moodle được thiết lập sử dụng port mặc định 80 với HTTP và 443 với HTTPS). Tiến hành cài đặt, cấu hình và tùy biến hệ thống LMS Moodle cho phù hợp với nhu cầu sử dụng.

#### 4.2. Triển khai hệ thống LMS Moodle trên môi trường đám mây AWS với Docker container

Phần tiếp theo là quá trình triển khai hệ thống LMS Moodle trên đám mây AWS với nền tảng container Docker.

- Bước 1: Tạo một instance AWS EC2: (1) Tải Image của máy ảo Amazon (AMI), (2) Lựa chọn

Ngày nhận bài: 15/4/2020

Ngày phản biện đánh giá và sửa chữa: 24/4/2020

Ngày chấp nhận đăng bài: 4/5/2020

*Thông tin tác giả:*

ThS. PHẠM NGỌC DUY

Giảng viên Khoa Công nghệ thông tin

Trường Đại học Hàng hải Việt Nam

**CONTAINERIZATION:  
A METHOD TO FAST DEPLOYING  
LEARNING MANAGEMENT SYSTEMS (LMS)  
ON THE CLOUD ENVIRONMENT**

● Master. PHAM NGOC DUY

Lecturer, Faculty of Information Technology,  
Vietnam Maritime University

**ABSTRACT:**

In the Industry 4.0 era, advanced information technology solutions should be adopted intensively by organizations and enterprises in all areas to effectively and quickly build and deploy management systems on non-traditional environments such as cloud computing. Learning Management Systems (LMS) are not only widely used in education sector but also are implemented by many corporations and enterprises to work out their learning and development strategies.

This paper presents an experimental implementation using Docker container platform and LMS Moodle for rapid deployment of LMS systems in cloud environment. Evaluations show that container technology provides a fast and efficient deployment of LMS in multiple platforms from on-premises to cloud.

**Keywords:** Containerization, Docker, cloud computing, Learning Management System.