



Research Article

สถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์

Architecture for Cloud-based Advanced Application Development Tools of Container Platform

อรนุช พันโท รสลิน เพตะกร * ภาณุวัฒน์ สุวรรณกุล และ กาญจนา ขัตติหะจักร์

Oranuch Pantho, Roselin Petagon *, Panuwat Suwannakul and Kanchana Khatthachak

ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเชียงใหม่ อำเภอเมือง จังหวัดเชียงใหม่ 50300
Computer Department, Faculty of Science and Technology, Chiang Mai Rajabhat University, Muang, Chiang Mai 50300, Thailand.

ABSTRACT

Article history:

Received: 2023-07-10

Revised: 2024-04-03

Accepted: 2024-05-07

Keywords:

tools;

containers;

advanced applications;

tool architecture;

cloud computing

Containers are an ideal tool for developing and deploying software. Organizations can leverage container architecture to enhance software development processes and more efficiently deliver high-performance applications. This research aims to 1) develop an architecture for advanced cloud-based application development using container platforms and 2) investigate user satisfaction with the developed architecture. The tools employed in this study include the architecture for cloud-based advanced application development using container platforms and satisfaction surveys regarding this architecture. The sample group was chosen through a specific purposive sampling technique and divided into two categories. The first category consisted of expert samples, including five computer science professors who specialize in the review of architectural tool design and development. The second category included 77 computer science students who were enrolled in Advanced Application Development course during the second and third academic years of 2022 at the Faculty of Science and Technology, Rajabhat Chiang Mai University. The analyzed data was collected using descriptive statistics, which included frequency, percentage, mean, and standard deviation.

The study results comprise two parts: 1) the architecture for advanced cloud-based application development using container platforms, which identifies three user groups within the system: administrators, system users (students), and instructors; and 2) the findings on student satisfaction with the cloud-based application development architecture using container platforms for cloud processing. The satisfaction level was high, with an average score of 4.37 and a standard deviation of 0.72.

* Corresponding author.

E-mail address: roselin@cmru.ac.th

Cite this article as:

Pantho, O., Petagon, R., Suwannakul, P. and Khatthachak, K. 2025. Architecture for Cloud-based Advanced Application Development Tools of Container Platform. *Recent Science and Technology* 17(2): 259694.

บทคัดย่อ

คอนเทนเนอร์เป็นเครื่องมือที่เหมาะสมสำหรับการพัฒนาและปรับใช้ซอฟต์แวร์ องค์กรต่าง ๆ สามารถใช้ประโยชน์จากสถาปัตยกรรมคอนเทนเนอร์เพื่อปรับปรุงกระบวนการพัฒนาซอฟต์แวร์และส่งมอบแอปพลิเคชันที่มีความสามารถสูงได้อย่างมีประสิทธิภาพมากขึ้น งานวิจัยนี้มีวัตถุประสงค์คือ 1) พัฒนาสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ด้วยแพลตฟอร์มคอนเทนเนอร์ และ 2) ศึกษาความพึงพอใจของผู้ใช้งานสถาปัตยกรรมที่พัฒนาขึ้น เครื่องมือที่ใช้ในการวิจัยได้แก่ สถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์และแบบสอบถามความพึงพอใจต่อสถาปัตยกรรมนี้ กลุ่มตัวอย่างเลือกโดยใช้วิธีการสุ่มตัวอย่างแบบเจาะจง แบ่งออกเป็น 2 กลุ่มคือ กลุ่มตัวอย่างผู้เชี่ยวชาญ คืออาจารย์ที่เป็นผู้เชี่ยวชาญด้านคอมพิวเตอร์เพื่อตรวจสอบการออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือ จำนวน 5 คน และ กลุ่มตัวอย่างคือ นักศึกษาภาควิชาคอมพิวเตอร์ที่ลงทะเบียนเรียนรายวิชาการพัฒนาแอปพลิเคชันขั้นสูง ในภาคเรียนที่ 2-3 ปีการศึกษา 2565 คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเชียงใหม่ จำนวน 77 คน การวิเคราะห์ข้อมูลโดยใช้สถิติเชิงพรรณนา ได้แก่ สถิติพื้นฐาน (ความถี่ ร้อยละ) ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน

ผลการศึกษาค้นนี้ประกอบด้วย 2 ส่วนคือ 1) สถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์โดยสถาปัตยกรรมมีผู้ใช้ระบบจำนวน 3 กลุ่มคือ ผู้ดูแลระบบ ผู้ใช้งานระบบ (ผู้เรียน) และผู้สอน และ 2) ผลการศึกษาความพึงพอใจของผู้เรียนต่อสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์โดยใช้คอนเทนเนอร์ของการประมวลผลคลาวด์ โดยผลการศึกษาความพึงพอใจอยู่ในระดับมาก ค่าเฉลี่ยคือ 4.37 และส่วนเบี่ยงเบนมาตรฐานคือ 0.72

คำสำคัญ: เครื่องมือ, คอนเทนเนอร์, แอปพลิเคชันขั้นสูง, สถาปัตยกรรมเครื่องมือ, การประมวลผลคลาวด์

1. บทนำ

คลาวด์หรือบางครั้งเรียกว่า การประมวลผลคลาวด์ (Cloud Computing) เป็นบริการระบบที่เป็นตัวกลางเพื่อให้ผู้ใช้ได้ใช้งานบริการต่าง ๆ ผ่านอินเทอร์เน็ต ในปัจจุบันนี้มีผู้ให้บริการคลาวด์ที่ให้เข้าใช้บริการต่าง ๆ บนคลาวด์หลายราย เช่น AWS (Amazon Inc., 2023), Google Cloud Platform (Google Inc., 2023), Azure (Microsoft, 2023) เป็นต้น ซึ่งการใช้บริการเหล่านี้จะต้องมีการเช่าทรัพยากรจากผู้ให้บริการอาจจะมีค่าใช้จ่ายสูงคิดค่าบริการต่อการใช้งานตามเซิร์ฟเวอร์ที่ผู้ใช้บริการทำการเปิดใช้งาน หากองค์กรนั้น ๆ ไม่สามารถเลือกเช่าทรัพยากรจากผู้ให้บริการได้ตรงกับความต้องการ ทำให้องค์กรอาจใช้ทรัพยากรเกินความจำเป็นได้ เพื่อลดค่าใช้จ่ายในการเช่าใช้บริการ Charoenphuthiwat and Buranasaksee (2023) จึงเสนอวิธีการพัฒนาเครื่องมือสร้างแบบจำลองบนแพลตฟอร์มคูเบอร์เนตส์ (Kubernetes) เพื่อช่วยให้องค์กรสามารถเลือกใช้ทรัพยากรที่เหมาะสมที่สุดและมีประสิทธิภาพสูงสุดในการใช้งานตามขอบเขตทรัพยากรที่คุ้มค่าที่สุดสำหรับแอปพลิเคชันให้สอดคล้องกับงบประมาณที่มีอยู่

ด็อกเกอร์ (Docker Inc., 2023) เป็นเครื่องมือที่เป็นที่รู้จักและเรียบง่ายที่สุดสำหรับการสร้างและใช้งานคอนเทนเนอร์ ช่วยให้สามารถพัฒนา ทดสอบบริการ และการตั้งค่าต่าง ๆ ได้อย่างราบรื่น ซึ่งคอนเทนเนอร์เป็นหน่วยน้ำหนักเบาและพกพาได้ง่าย ซึ่งใช้ในการรวมแอปพลิเคชันเข้ากับการตั้งค่าและไลบรารีทั้งหมด โดยให้ความสอดคล้องและความยืดหยุ่นบนแพลตฟอร์มที่

หลากหลาย คอนเทนเนอร์จึงเป็นเครื่องมือสำหรับการพัฒนาสถาปัตยกรรมไมโครเซอร์วิส เนื่องจากมีกรอบการทำงานที่สามารถแยกบริการต่าง ๆ ได้ โดยแต่ละบริการมีฟังก์ชันการทำงานของตัวเอง ในขณะที่ยังคงมีการประสานงานอย่างเหมาะสม รวมทั้งช่วยในการย้ายแอปพลิเคชันและการปรับปรุงรุ่นแอปพลิเคชันได้สะดวก รวดเร็ว เทคโนโลยีสำหรับการพัฒนาสถาปัตยกรรมไมโครเซอร์วิส เช่น ด็อกเกอร์ และ คูเบอร์เนตส์

การศึกษาในปัจจุบันจะเน้นผู้เรียนเป็นสำคัญ โดยให้ผู้เรียนได้ลงมือปฏิบัติด้วยตนเอง แสวงหาความรู้จากแหล่งเรียนรู้และวิธีการเรียนรู้ที่หลากหลายในการพัฒนาแอปพลิเคชันต้องทำการติดตั้งซอฟต์แวร์ในเครื่องคอมพิวเตอร์ตนเอง การติดตั้งซอฟต์แวร์ที่จำเป็นต้องใช้ในการพัฒนาสำหรับแอปพลิเคชันขั้นสูงนั้นต้องคำนึงถึงเครื่องมือที่ใช้พัฒนา สภาพแวดล้อมทางด้านฮาร์ดแวร์ และซอฟต์แวร์ที่ติดตั้งสำหรับการพัฒนาแอปพลิเคชันขั้นสูง ซึ่งต้องมีความชำนาญพอสมควรในการติดตั้งสภาพแวดล้อมสำหรับการพัฒนา ในบางครั้งการเริ่มต้นพัฒนาเว็บแอปพลิเคชันขั้นสูงจุดเริ่มต้นคือการติดตั้งซอฟต์แวร์ที่เหมาะสมในเครื่องคอมพิวเตอร์ที่จะใช้พัฒนา เมื่อไม่สามารถติดตั้งเครื่องมือที่หลากหลายเหล่านั้นได้สำเร็จก็ไม่สามารถเริ่มต้นพัฒนาแอปพลิเคชันขั้นสูงได้เช่นเดียวกัน ในปัจจุบันมีการจัดการเรียนการสอนแบบเอ็มเลิร์นนิ่งด้วยเทคโนโลยีการคำนวณแบบคลาวด์ (Laisema, 2018) เป็นการจัดการเรียนโดยให้ผู้เรียนเรียนรู้ได้ทุกที่ผ่านอุปกรณ์พกพาที่มีการเชื่อมต่อผ่านระบบเครือข่าย โดยการเรียนรู้การเขียนโปรแกรมภาษาต่าง ๆ

ในปัจจุบันมีเครื่องมือที่สามารถใช้เขียนโปรแกรมภาษาต่าง ๆ ได้ผ่านเว็บเบราว์เซอร์ของผู้ให้บริการที่เป็นที่นิยมคือ เวอร์ชวลสตูดิโอโค้ด (Visual Studio Code) ผ่านเว็บ (Microsoft Inc., 2023) ในการใช้งานผ่านเบราว์เซอร์เทอร์มินัลจะไม่พร้อมให้ใช้งานในตัวแก้ไขเว็บ แต่ถ้าต้องการใช้งานเทอร์มินัลให้ใช้งานผ่าน Codespace หรือ VS Code Desktop เท่านั้น และมีบริษัท Gitpod GmbH (2023) ที่ให้บริการสภาพแวดล้อมสำหรับการพัฒนาแอปพลิเคชันเช่นกันเปิดให้ใช้ฟรีเฉพาะฟังก์ชันการทำงานพื้นฐานหากต้องการฟังก์ชันการทำงานขั้นสูงก็มีการคิดค่าใช้จ่ายการใช้งานเป็นรายชั่วโมงต่อการพัฒนาแอปพลิเคชัน

การพัฒนาแอปพลิเคชันขั้นสูงเป็นรายวิชาที่จะเกี่ยวกับการออกแบบส่วนติดต่อกับผู้ใช้งาน ซึ่งเป็นส่วนหน้าบ้าน (Front End) ที่สามารถทำงานได้นับอุปกรณ์ที่หลากหลายหน้าจอที่แตกต่างกันก็สามารถแสดงผลได้ผ่านเว็บเบราว์เซอร์ ส่วนหลังบ้าน (Back End) ที่ใช้ในการติดต่อและจัดการฐานข้อมูลสำหรับการพัฒนาแอปพลิเคชันซึ่งผู้เรียนจะต้องมีทรัพยากรมากพอสำหรับติดตั้งเครื่องมือสำหรับการพัฒนา และมีเวลาสำหรับฝึกปฏิบัติอย่างสม่ำเสมอ

ดังนั้น ผู้วิจัยจึงมีแนวคิดที่จะพัฒนาสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ที่มีความยืดหยุ่นโดยได้แบ่งปันทรัพยากรต่าง ๆ ผ่านเครือข่ายอินเทอร์เน็ต ให้ผู้ใช้งานสถาปัตยกรรมเครื่องมือผ่านเว็บเบราว์เซอร์เพื่อพัฒนาแอปพลิเคชันขั้นสูงซึ่งมีส่วนประกอบ 3 ส่วนสำหรับผู้เรียนคือส่วนที่เป็นขั้นตอนการติดตั้งและพัฒนาแอปพลิเคชันให้ผู้เรียนเรียนรู้ทีละขั้นตอนโดยการคลิกขั้นตอนต่าง ๆ ใน

ส่วนที่ 2 เครื่องมือจะทำงานตามขั้นตอนให้อัตโนมัติซึ่งคือเวอร์ชวลสตูดิโอโค้ดพร้อมทั้งเปิดใช้งานเทอร์มินัลให้ผ่านเบราว์เซอร์ และส่วนที่ 3 ส่วนเบราว์เซอร์ก็เก็กลิงก์และส่วนฐานข้อมูลโดยผู้เรียนได้คอนเทนเนอร์ที่สามารถเข้าถึงทรัพยากรในการพัฒนาแอปพลิเคชันขั้นสูงทั้งส่วนหน้าบ้าน หลังบ้านและฐานข้อมูล ผู้สอนสามารถบริหารจัดการคอนเทนเนอร์โดยมีระบบจัดการเนื้อหาเพื่อช่วยสร้างเนื้อหาในการจัดการเรียนการสอนแบบปฏิบัติการให้ผู้เรียนสามารถเรียนรู้นอกห้องเรียนได้ด้วยตนเองซ้ำก็รอบก็ได้ด้วยตัวผู้เรียนเอง ผู้ดูแลระบบสามารถจัดการคอนเทนเนอร์ทั้งหมด สิ่งสำคัญคือผู้เรียนมีพื้นที่ในการเก็บโค้ดที่ได้พัฒนาขึ้นเป็นประวัติการพัฒนาโค้ดสำหรับนำไปใช้ต่อยอดและใช้ในการสมัครงานต่อไปในอนาคตได้

งานวิจัยนี้มีวัตถุประสงค์ 1) เพื่อออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ และ 2) เพื่อศึกษาความพึงพอใจของผู้ใช้งานของสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์

2. วิธีดำเนินการวิจัย

งานวิจัยนี้มีวิธีการดำเนินการวิจัยดัง Figure 1 ประกอบด้วย 5 ขั้นตอน คือ กลุ่มเป้าหมาย เครื่องมือที่ใช้ในการวิจัย การวิเคราะห์และสังเคราะห์แนวคิดสถาปัตยกรรมเครื่องมือ การออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือ และการวิเคราะห์ผล โดยมีรายละเอียดดังต่อไปนี้

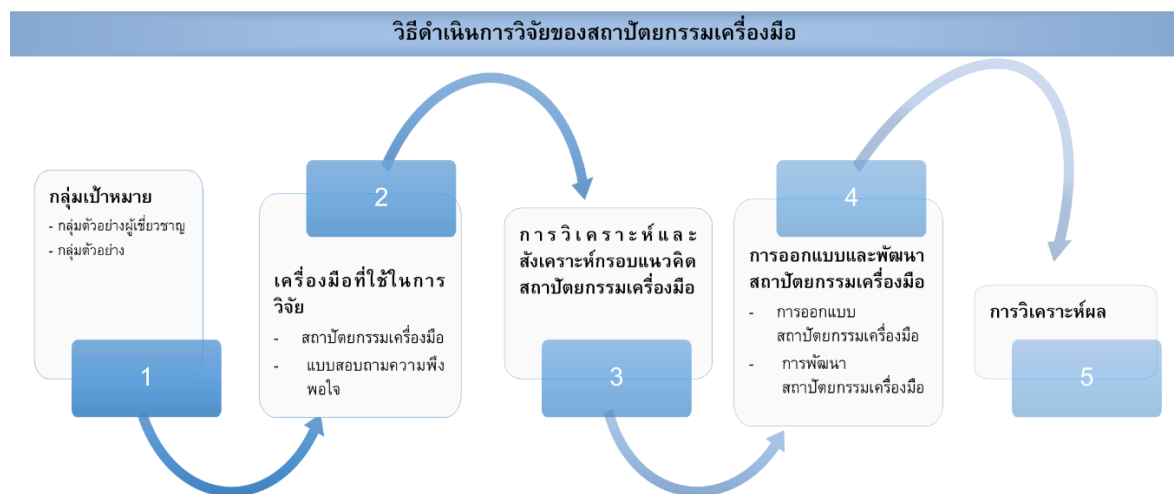


Figure 1 Research methods of the Architecture Tool

2.1 กลุ่มเป้าหมาย

ประชากร ได้แก่ อาจารย์และนักศึกษามหาวิทยาลัยราชภัฏเชียงใหม่

กลุ่มตัวอย่าง ผู้วิจัยได้เลือกวิธีการสุ่มตัวอย่างแบบเจาะจง (Purposive Sampling) (Wanichbancha, 2011) โดยเลือกศึกษาจากประชากรที่มีลักษณะตรงตามวัตถุประสงค์ที่จะศึกษา เพื่อให้ได้บุคคลที่สามารถให้ข้อมูลตรงตามที่ต้องการ โดยแบ่งออกเป็น 2 กลุ่มคือ

- กลุ่มตัวอย่างผู้เชี่ยวชาญ คืออาจารย์ที่เป็นผู้เชี่ยวชาญด้านคอมพิวเตอร์เพื่อตรวจสอบการออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือ จำนวน 5 คน
- กลุ่มตัวอย่าง คือนักศึกษาภาควิชาคอมพิวเตอร์ที่ลงทะเบียนเรียนรายวิชา การพัฒนาแอปพลิเคชันขั้นสูงในภาคเรียนที่ 2-3 ปีการศึกษา 2565 คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเชียงใหม่ จำนวน 77 คน

2.2 เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ใช้ในการวิจัย ได้แก่

- 1) สถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์
- 2) แบบสอบถามความพึงพอใจในความพึงพอใจต่อสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น

2.3 การวิเคราะห์และสังเคราะห์กรอบแนวคิดสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น

ผู้วิจัยได้ทำการศึกษาทฤษฎี บทความ และงานวิจัยที่เกี่ยวข้องกับแนวทางในการออกแบบสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ เกี่ยวกับการใช้เทคโนโลยีคอนเทนเนอร์ของการประมวลผลคลาวด์เพื่อให้การพัฒนาแอปพลิเคชันที่นำไปใช้จริงในองค์กรสามารถทำการติดตั้งและปรับขนาดของแอปพลิเคชันให้เหมาะสมกับ

ทุกสภาพแวดล้อม และพร้อมใช้งาน ทำให้ผู้ใช้งานแอปพลิเคชันสามารถเรียกใช้งานได้อย่างรวดเร็ว ช่วยให้สามารถพัฒนาแอปพลิเคชันขั้นสูงได้อย่างรวดเร็วเช่นเดียวกัน พร้อมทั้งยังสามารถส่งมอบแอปพลิเคชันที่ทันสมัยไปยังระบบคลาวด์ได้เร็วยิ่งขึ้น

2.4 การออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น

การออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์โดยมีรายละเอียด ดังต่อไปนี้

2.4.1 การออกแบบสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์

การออกแบบและพัฒนาสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นบนพื้นฐานเทคนิควงจรชีวิตการพัฒนาซอฟต์แวร์ (Software Development Life Cycle: SDLC) (Gary and Harry, 2011) ซึ่งนำมาใช้ในการออกแบบยูสเคสโคอะแกรมของสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ให้แสดงสถาปัตยกรรมที่มีผู้ใช้งานที่เกี่ยวข้องในระบบจำนวน 3 ส่วนได้แก่ ผู้ดูแลระบบ ผู้ใช้งานระบบ (ผู้เรียน) และผู้สอน การออกแบบสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นเน้นใช้เครื่องมือช่วยในการจัดการเรียนการสอนให้ลดความยุ่งยากในการติดตั้งเครื่องมือต่าง ๆ ในการพัฒนาแอปพลิเคชันขั้นสูง ซึ่งสถาปัตยกรรมนี้จะเป็นเครื่องมือให้ผู้สอนสามารถเตรียมเนื้อหาบทเรียนให้ผู้เรียนเพื่อสามารถเรียนตามลำดับขั้นตอนเพื่อให้เข้าใจง่าย พร้อมทั้งสามารถทบทวนเนื้อหาที่ได้เรียนรู้ด้วยตัวเองได้อีกด้วยสำหรับผู้ดูแลระบบสามารถค้นหาข้อมูลผู้ใช้ และจัดการผู้ใช้ และผู้ใช้งานระบบที่เป็นผู้เรียนสามารถเข้าไปเรียนรู้ได้พร้อม ๆ ผู้สอน และทบทวนเนื้อหาที่ได้เรียนรู้ด้วยตนเองได้ตลอดเวลาดังใน Figure 2

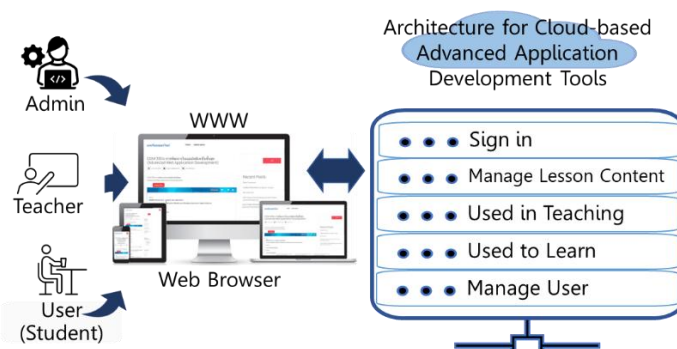


Figure 2 Concept of the Architecture Tool

การออกแบบสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ ทำให้ได้แนวทางในการออกแบบยูสเคสไดอะแกรม ดังใน Figure 3 ที่ประกอบไปด้วยผู้ดูแลระบบ (Admin) ผู้สอน (Teacher) และผู้ใช้งานหรือผู้เรียน (Student) โดยที่ผู้ดูแลระบบเข้าสู่ระบบเพื่อจัดการผู้สอน และผู้เรียน ในส่วนของผู้สอนเข้าสู่ระบบเพื่อเตรียมเนื้อหาในการสอนเป็นขั้นตอนสำหรับการฝึกปฏิบัติในห้องเรียนและผู้เรียนสามารถทบทวนได้หลังจากที่ได้ฝึกปฏิบัติในห้องเรียนเรียบร้อยแล้ว เมื่อเกิดปัญหาระหว่างที่ผู้เรียนฝึกปฏิบัติผู้สอนหรือผู้ดูแลระบบสามารถเข้าไปในคอนเทนเนอร์ของผู้เรียน ผู้สอนกับผู้เรียนจะได้หน้าตาเดียวกันแล้วช่วยแก้ปัญหาได้ทันที สำหรับผู้เรียนเมื่อสมัครเข้าใช้งานระบบ ระบบจะทำการสร้างคอนเทนเนอร์ที่เหมาะสมสำหรับการเรียน ในรายวิชาการพัฒนาแอปพลิเคชันขั้นสูง ให้ผู้เรียนสามารถเรียนรู้ได้ตามขั้นตอนที่มีให้จากผู้สอนได้สร้างขึ้น และผู้เรียนสามารถเรียนรู้เมื่อไหร่ก็ได้ตามความต้องการ

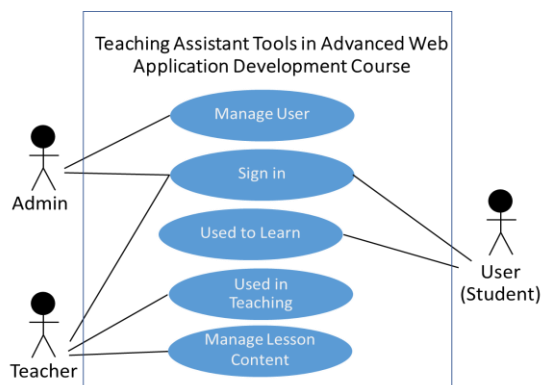


Figure 3 The Use Case Diagram of Teaching Assistant Tool Architecture

2.4.2 การพัฒนาสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น

การพัฒนาสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นทำให้ผู้สอนสามารถใช้สถาปัตยกรรมเครื่องมือในรายวิชาการพัฒนาแอปพลิเคชันขั้นสูง ผู้เรียนสามารถใช้ระบบบนคอนเทนเนอร์ของผู้เรียนเองเพื่อใช้ในการเรียนสำหรับพัฒนาเว็บแอปพลิเคชันโดยไม่จำเป็นต้องติดตั้งเครื่องมือต่าง ๆ ในการพัฒนา ระบบได้เตรียมสภาพแวดล้อมในการพัฒนาให้พร้อมสำหรับเริ่มพัฒนาเว็บแอปพลิเคชัน พร้อมทั้งสามารถเก็บโค้ดในการพัฒนาโปรแกรมไว้ในคอนเทนเนอร์ส่วนตัว และส่งลิงค์ผลลัพธ์ที่ได้พัฒนาให้กับผู้ใช้อื่นทดสอบระบบได้โดยไม่ต้องทำการเข้าสู่ระบบนั้นคือเหมือนเว็บแอปพลิเคชันถูกพัฒนาแล้วย้ายไปอยู่บนเครื่องให้บริการแม่ข่ายเรียบร้อยแล้ว และผู้สอนสามารถเตรียม

เนื้อหาบทเรียนให้กับผู้เรียนได้ พร้อมทั้งมีการประเมินประสิทธิภาพการพัฒนาระบบเพื่อให้สามารถใช้งานระบบได้ดียิ่งขึ้น ผู้ใช้ระบบสามารถใช้งานสถาปัตยกรรมเครื่องมือได้ผ่านเว็บเบราว์เซอร์บนอุปกรณ์เคลื่อนที่หรือบนเครื่องคอมพิวเตอร์ก็ได้ที่เชื่อมต่ออินเทอร์เน็ตผู้ใช้ก็สามารถเข้าใช้งานสถาปัตยกรรมเครื่องมือนี้ได้ ซึ่งเป็นการให้บริการด้านซอฟต์แวร์ (Software as a Service) ในรูปแบบหนึ่งของการประมวลผลคลาวด์ โดยผู้ใช้สามารถใช้งานได้พร้อม ๆ กันในเวลาเดียวกันได้อีกด้วย (Mell and Grance, 2011)

สถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น มีการจัดการเนื้อหาด้วยภาษาพีเอชพี ภาษาจาวาใช้จัดการการลงทะเบียน มีการจัดเก็บข้อมูลในฐานข้อมูล MySQL ด้วยโปรแกรม XAMPP และสร้างคอนเทนเนอร์สำหรับผู้ใช้งานหรือผู้เรียนด้วย Docker ที่สามารถจัดการได้ผ่าน NGINX ดังแสดงใน Figure 4 เป็นหน้าตาที่แสดงรายละเอียดเกี่ยวกับรายวิชา มีรายละเอียดของเนื้อหาเบื้องต้นให้ผู้เรียนให้เป็นภาพรวมของเนื้อหาทั้งหมด และผู้เรียนสามารถกดปุ่มเข้าสู่บทเรียนเพื่อสมัครเข้าเรียนในระบบระบบจะทำการสร้างคอนเทนเนอร์ให้ผู้เรียนแต่ละคนเข้าในระบบจะปรากฏ ดังใน Figure 6 ซึ่งผลลัพธ์ในการพัฒนาสถาปัตยกรรมเครื่องมือประกอบด้วยส่วนต่าง ๆ ดังนี้

- ส่วนผู้ใช้งานระบบในฐานะผู้เรียนและผู้สอน ดังใน Figure 6 เมื่อระบบทำการสร้างคอนเทนเนอร์ให้เรียบร้อยแล้วผู้ใช้งานจะได้หน้าตาที่มีส่วนประกอบ 3 ส่วนคือซ้ายมือเป็นเมนูสำหรับขั้นตอนการเรียนที่ผู้สอนสร้างขึ้นมาเป็นเนื้อหาบทเรียนในแต่ละบท โดยเนื้อหาประกอบด้วยขั้นตอนการพัฒนาประกอบด้วยคำสั่งสำหรับการติดตั้งสภาพแวดล้อมและคำสั่งในการพัฒนาแอปพลิเคชันขั้นสูงที่เป็นแนวทางให้ผู้เรียนสามารถเรียนรู้ด้วยตนเอง ผู้สอนสามารถใช้ในการสอนตามขั้นตอนโดยที่ผู้เรียนหรือผู้สอนไม่ต้องพิมพ์แค่คำสั่งทางด้านซ้ายบางคำสั่งที่ต้องการทำงานผ่านเทอร์มินอลก็ทำการเปิดเทอร์มินอลเพื่อทำงาน บางคำสั่งต้องการทำงานในส่วนใดก็ทำการเปิดส่วนนั้นขึ้นมาทำงานได้แบบอัตโนมัติ ตรงกลางเป็นหน้าต่างเครื่องมือสำหรับพัฒนาแอปพลิเคชัน Visual code Studio และส่วนขวามือเป็นหน้าต่างเว็บเบราว์เซอร์แสดงผลของการพัฒนาหน้าเว็บแอปพลิเคชันขั้นสูงแบบทันทีทันใดเมื่อผู้ใช้ทำการรันผลลัพธ์การทำงานไว้ ในส่วนของผู้สอนสามารถจัดการหน้าตา รวมทั้งหน้าตาฐานข้อมูลต่าง ๆ

- ส่วนผู้สอนสำหรับจัดการเนื้อหา ดังใน Figure 5 จะปรากฏหน้าตาที่ผู้สอนสามารถสร้างขั้นตอนหรือเนื้อหาสำหรับการสอนได้ทางซ้ายมือซึ่งเนื้อหาจะประกอบด้วยคำสั่งที่อยู่ในรูปแบบความต้องการให้รันหรือต้องการให้

คัดลอกเพื่อแก้ปัญหาเมื่อผู้สอนแล้วผู้เรียนทำตามแล้วพิมพ์ซ้ำหรือพิมพ์ผิดทำให้เกิดปัญหาการเรียนไม่ทัน และทางซ้ายจะเป็นผลลัพธ์ที่แสดงผลให้ผู้เรียนจริงในหน้าต่างซ้ายมือ ดังใน Figure 6

- ส่วนผู้ดูแลระบบ ที่สามารถจัดการค้นหา ลบ แก้ไขข้อมูล และจัดการสิทธิ์การใช้งาน ดังใน Figure 7 เป็น

ตัวอย่างของหน้าต่างที่ผู้ดูแลระบบเห็นข้อมูลผู้ใช้ระบบทั้งหมดว่าผู้ใช้ระบบได้คอนเทนต์ไหน ตอนนี้คอนเทนต์ทำงานอยู่หรือไม่ หากไม่ต้องการใช้งานคอนเทนต์แล้ว จะทำการหยุดคอนเทนต์และลบคอนเทนต์นั้นทิ้งได้ หรือถ้าหากผู้ใช้ลืมรหัสผ่านผู้ดูแลระบบก็จะสามารถจัดการให้ได้

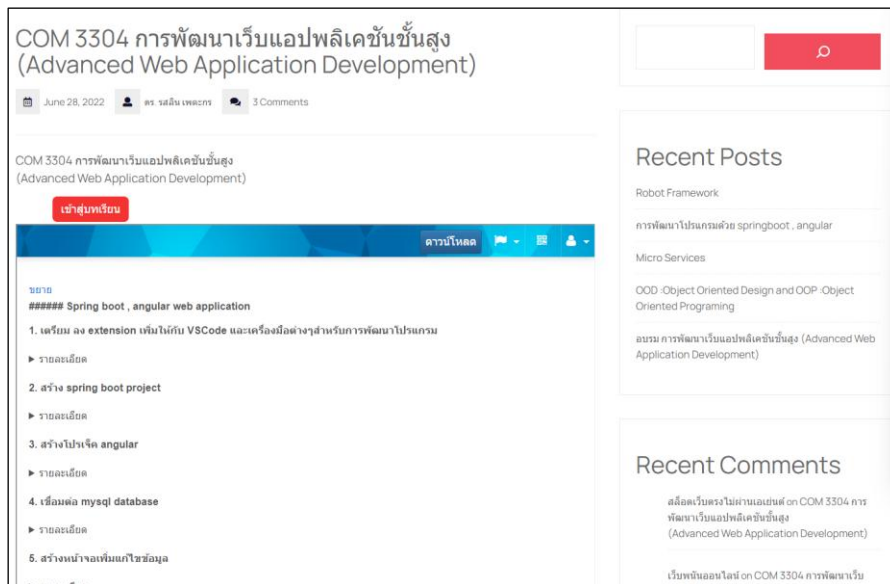


Figure 4 Home Page of Teaching Assistant Tools

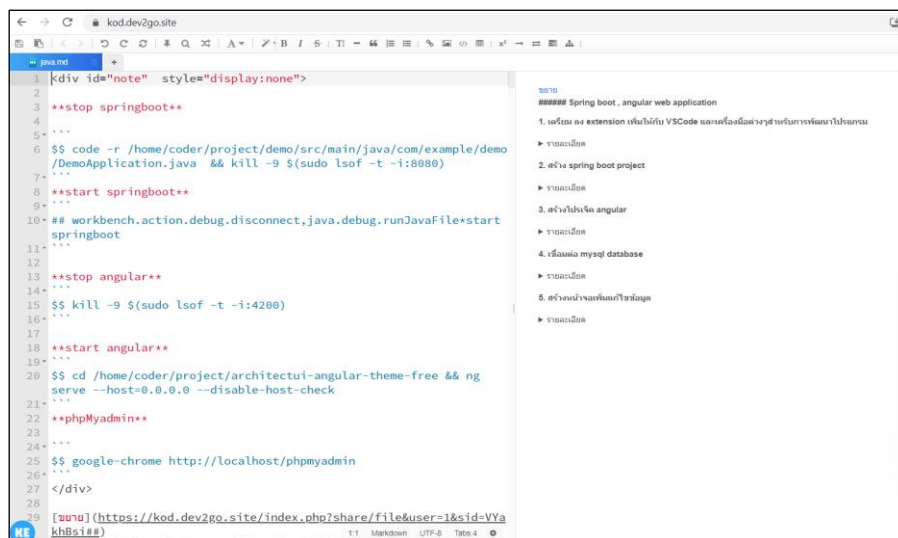


Figure 5 Content Management Page of the Teacher

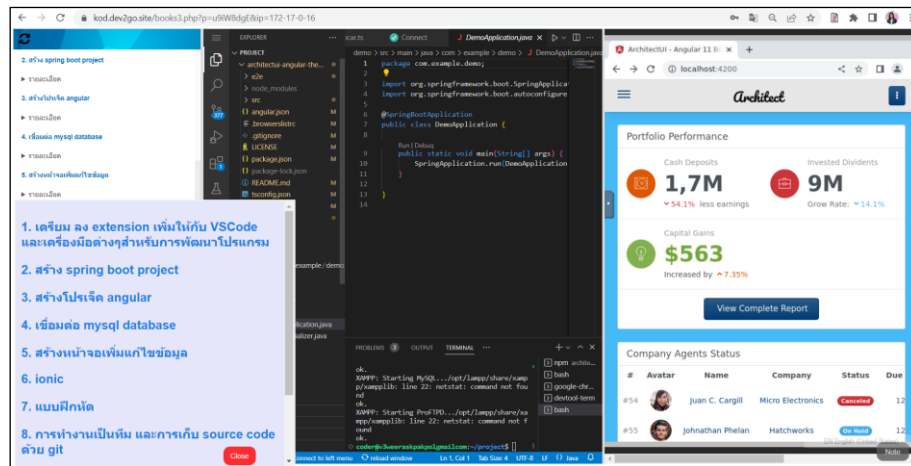


Figure 6 Learning of Students and Teaching of Teachers Pages

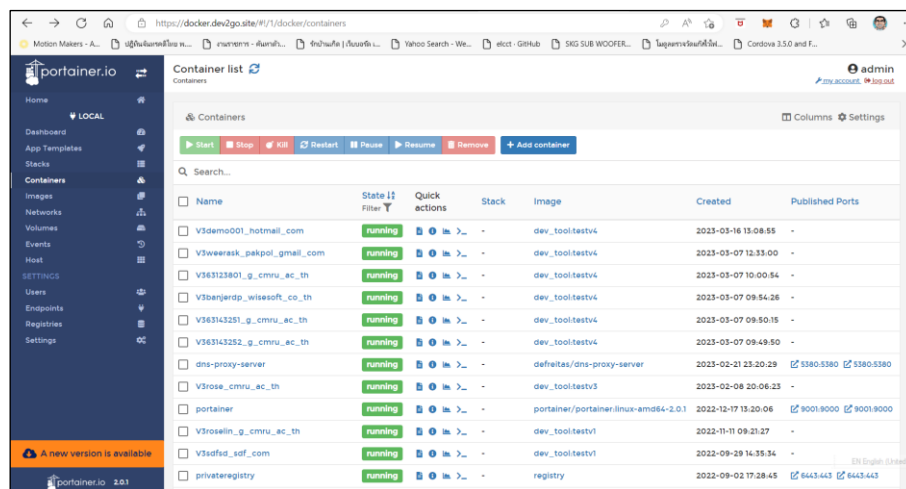


Figure 7 Management Page of Admin

2.5 การวิเคราะห์ผล

ผู้วิจัยวิเคราะห์ข้อมูลโดยใช้สถิติเชิงบรรยาย ได้แก่ ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน แล้วนำผลค่าเฉลี่ยมาเปรียบเทียบกับเกณฑ์การแปลความหมายตามเกณฑ์ (Srisa-ard, 2017) ดังนี้

- ค่าเฉลี่ย 4.50 – 5.00 เท่ากับระดับ มากที่สุด
- ค่าเฉลี่ย 3.50 – 4.49 เท่ากับระดับ มาก
- ค่าเฉลี่ย 2.50 – 3.49 เท่ากับระดับ ปานกลาง
- ค่าเฉลี่ย 1.50 – 2.49 เท่ากับระดับ น้อย
- ค่าเฉลี่ย 1.00 – 1.49 เท่ากับระดับ น้อยที่สุด

3. ผลการวิจัยและวิจารณ์ผล

ผลการวิเคราะห์ความพึงพอใจสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ ประกอบด้วย 3 ส่วนคือ ภาพรวมของสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น ผลการ

วิเคราะห์ความพึงพอใจของผู้เชี่ยวชาญ และผลการวิเคราะห์ความพึงพอใจของผู้ใช้

3.1 ภาพรวมของสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้น ใน Figure 8 สถาปัตยกรรมเครื่องมือต้องการนำเสนอการใช้เทคโนโลยีคอนเทนเนอร์และติดตั้ง Portainer เพื่อจัดการคอนเทนเนอร์ทั้งหมดที่มีในส่วนผู้ดูแลระบบ เมื่อผู้ใช้สมัครเข้าทำงานผู้ใช้จะได้โนดที่เชื่อมต่อกับส่วนเก็บข้อมูลที่อนุญาตให้เก็บข้อมูลในโดเมน Project เพื่อใช้เก็บข้อมูลสำหรับแต่ละโนด โดยแต่ละโนดจะทำการติดตั้งเครื่องมือที่จำเป็นต้องใช้หรือเครื่องมือพื้นฐานให้ผู้ใช้สามารถใช้งานระบบได้ นั่นคืออิมเมจสำหรับการพัฒนาแอปพลิเคชันขั้นสูง โดยทุก ๆ คอนเทนเนอร์รันด็อกเกอร์สคริปตามขั้นตอนสคริปต์ในด็อกเกอร์ไฟล์ (Dockerfile) ดังใน Figure 9 ที่เก็บคำสั่งสำหรับรันด็อกเกอร์สคริปเพื่อสร้างอิมเมจที่เป็นต้นแบบใน Figure 10 ในการสร้างคอนเทนเนอร์ที่สามารถใช้ในการพัฒนาแอปพลิเคชันขั้นสูงได้ เมื่อรันด็อกเกอร์คอนเทนเนอร์ตามสถาปัตยกรรมเครื่องมือที่ออกแบบ ผู้ใช้จะได้คอนเทนเนอร์

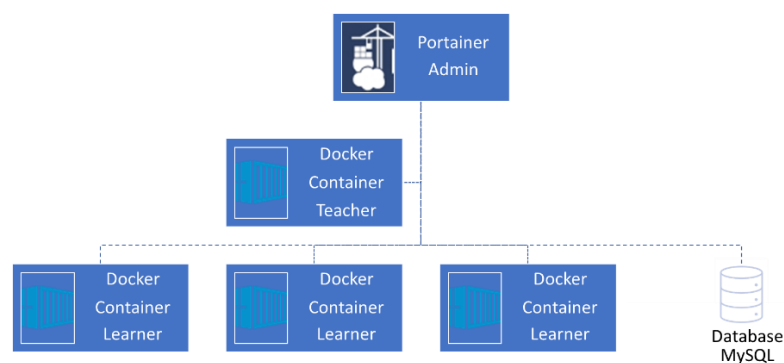


Figure 8 Overview of Architecture Tool

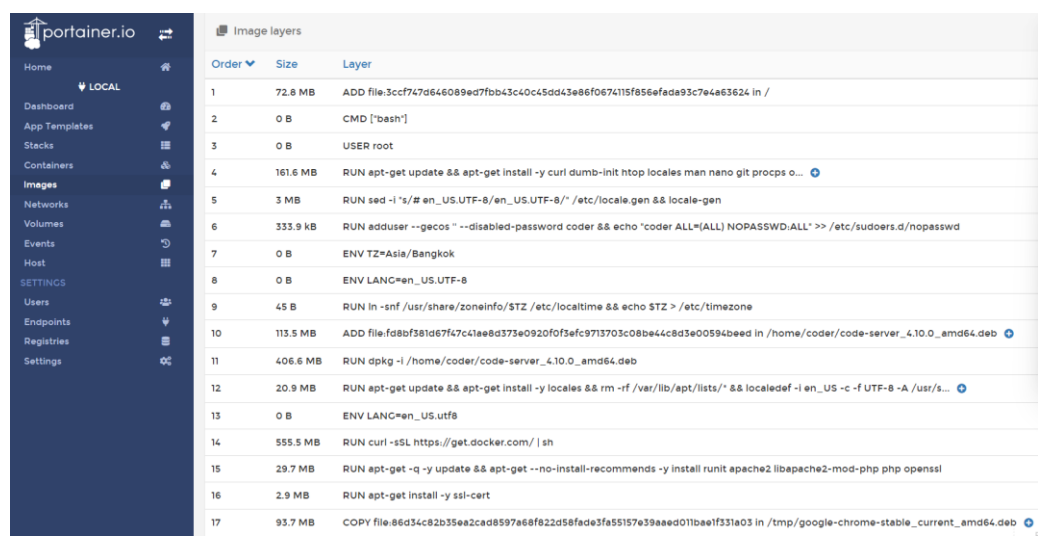


Figure 9 Docker Script for Install Container

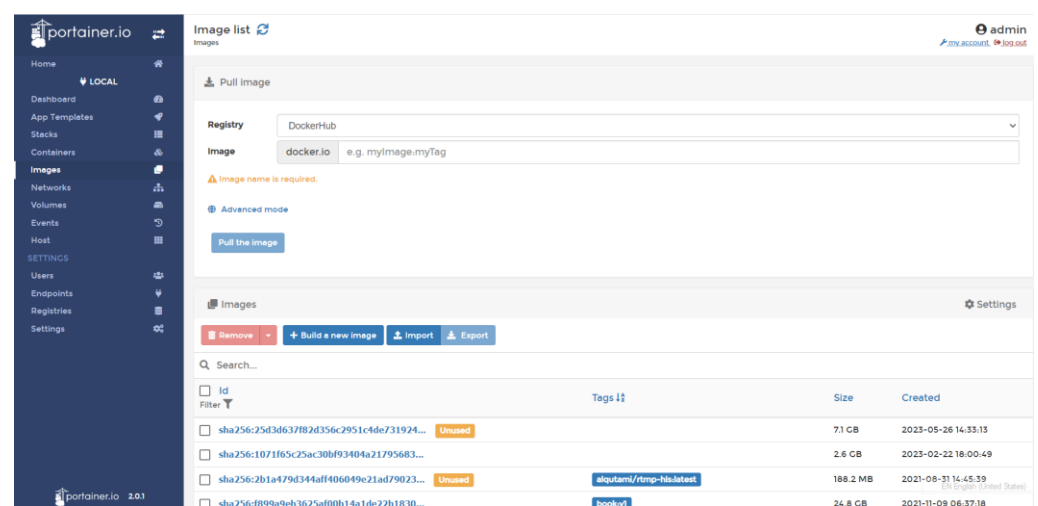


Figure 10 Container Image for Advanced Application Development

Table 1 The results of the opinion analysis of experts on the developed architecture (N=5)

Evaluation List	Evaluation Result		Interpretation
	$\bar{X}$	S.D.	
Function Requirement Test	4.50	0.353	Excellent
Function Test	4.52	0.672	Excellent
Usability Test	4.32	0.540	Good
Security Test	4.28	0.228	Good
Performance	4.55	0.512	Excellent
Teaching and Learning	4.53	0.447	Excellent
Total Average	4.45	0.458	Good

3.2 ผลการวิเคราะห์ความพึงพอใจของผู้เชี่ยวชาญต่อสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ จากผู้เชี่ยวชาญ จำนวน 5 คน ได้เลือกวิธีการสุ่มตัวอย่างแบบเจาะจง (Purposive Sampling) (Wanichbancha, 2011) ด้วยการใช้แบบประเมินที่เป็นแบบประมาณค่า 5 ระดับ ความพึงพอใจโดยภาพรวมแสดงดัง Table 1

จาก Table 1 ความคิดเห็นของผู้เชี่ยวชาญที่มีต่อสถาปัตยกรรมที่พัฒนาขึ้น พบว่า ความมีประสิทธิภาพของระบบมากที่สุด ค่าเฉลี่ย 4.55 ส่วนเบี่ยงเบนมาตรฐานเท่ากับ 0.512 ด้านงานการเรียนการสอนค่าเฉลี่ย 4.53 ส่วนเบี่ยงเบนมาตรฐาน 0.447 ด้านฟังก์ชันการทำงานได้ตามหน้าที่ค่าเฉลี่ย 4.52 ส่วนเบี่ยงเบนมาตรฐาน 0.672

และด้านตรงตามความต้องการมีค่าเฉลี่ย 4.50 ส่วนเบี่ยงเบนมาตรฐาน 0.353 ซึ่งโดยภาพรวมแล้วผู้เชี่ยวชาญเห็นว่าระบบมีค่าเฉลี่ยรวม 4.45 ส่วนเบี่ยงเบนมาตรฐาน 0.458 อยู่ในระดับมากสอดคล้องกับงานวิจัยของ (Bamroongcheep, 2020) ที่ได้พัฒนารูปแบบห้องเรียนกลับทางผ่านกูเกิ้ลคลาวด์คอมพิวติ้งและสื่อสังคมเพื่อส่งเสริมทักษะวิถีแห่งการคิดในศตวรรษที่ 21 สำหรับนิสิตครู ซึ่งผลความพึงพอใจโดยรวมของกลุ่มทดลองอยู่ในระดับมากและผู้เชี่ยวชาญรับรองรูปแบบที่พัฒนาขึ้นเห็นว่าภาพรวมมีความเหมาะสมมากที่สุด

3.3 ผลการวิเคราะห์ข้อมูลทั่วไปของผู้ใช้ที่ตอบแบบสอบถามความพึงพอใจต่อสถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์แสดงดังนี้

Table 2 The general data analysis results of users on the developed architecture (N=77)

List	Number	Percentage
Gender		
Male	55	71.43
Female	22	28.57
Total	77	100.00

Table 3 The results of the opinion analysis of user on the developed architecture (N=77)

Evaluation List	Evaluation Result		Interpretation
	$\bar{X}$	S.D.	
Function Requirement Test			
Functional Requirement Test on the Use Side	4.32	0.715	Good
Function Requirement Test on Coding	4.31	0.782	Good
Function Requirement Test on the output of a running program	4.29	0.792	Good
Function Requirement Test the correct display output and requirement	4.21	0.848	Good
Average	4.28	0.784	Good

Table 3 (Continuous)

Evaluation List	Evaluation Result		Interpretation
	$\bar{x}$	S.D.	
Function Test			
Function Test has the correct method of logging in to the system.	4.40	0.674	Good
The function test is correct when running the program.	4.35	0.739	Good
Function testing measures the accuracy of the program code.	4.42	0.676	Good
Function Test is correct in running the program.	4.39	0.728	Good
The function test has the accuracy of displaying the results.	4.44	0.678	Good
Average	4.40	0.699	Good
Usability Test			
The usability test is easy to run on the system.	4.27	0.789	Good
The Usability Test has a screen design that works properly.	4.29	0.758	Good
Usability Test The clarity of the text displayed on the screen	4.40	0.765	Good
The usability test is convenient to run.	4.25	0.746	Good
Usability Test The system is reliable and usable.	4.40	0.693	Good
Average	4.32	0.750	Good
Security Test			
The security test is licensed prior to the use of the various systems involved.	4.45	0.660	Good
Security tests are safe ways to verify information.	4.44	0.716	Good
Security Test is always giving advice and helping.	4.52	0.641	Excellent
Average	4.47	0.672	Good
Performance			
Performance The speed of accessing the service system	4.14	0.854	Good
Performance: Program processing speed	4.25	0.830	Good
Performance recording speed and program improvements	4.25	0.797	Good
Performance The speed of presenting results	4.31	0.782	Good
Performance The overall speed of the system	4.30	0.708	Good
Average	4.25	0.794	Good

จาก Table 3 มีการวิเคราะห์ความคิดเห็นของผู้ใช้ทั้งหมด 6 ด้าน ซึ่งพบว่า ด้านงานการเรียนการสอนมีความพึงพอใจมากที่สุด ค่าเฉลี่ย 4.54 ส่วนเบี่ยงเบนมาตรฐาน 0.634 รองลงมาเป็นด้านการรักษาความปลอดภัยของข้อมูล ค่าเฉลี่ย 4.47 ส่วนเบี่ยงเบนมาตรฐาน 0.672 และด้านการทำงานตามหน้าที่ ค่าเฉลี่ย 4.40 ส่วนเบี่ยงเบนมาตรฐาน 0.699 สำหรับผลของค่าเฉลี่ยรวมทุกด้านอยู่ในระดับมากค่าเฉลี่ย 4.37 ส่วนเบี่ยงเบนมาตรฐาน 0.722 อาจจะเนื่องจากเป็นระบบที่ช่วยลดจำนวนทรัพยากรในเครื่องของผู้ใช้และสามารถเข้าใช้งานได้ทุกที่ทุกเวลาสอดคล้องกับงานวิจัยของ Thitipetchakul *et al.* (2020) ที่ได้พัฒนารูปแบบการจัดการเรียนรู้บนระบบคลาวด์

คอมพิวเตอร์ตามแนวคิดคอนเนคตวิสต์ซิมเพื่อส่งเสริมการเรียนรู้เทคโนโลยีสารสนเทศและการสื่อสาร สำหรับนักศึกษาระดับปริญญาตรี พบว่าผู้เรียนรู้ระบบคลาวด์คอมพิวเตอร์ที่พัฒนาขึ้นอยู่ในระดับมากที่สุด และงานวิจัยของ Wannasin *et al.* (2021) ได้วิจัยผลการจัดกิจกรรมการเรียนรู้เชิงรุกโดยใช้ปัญหาเป็นฐานผ่านเทคโนโลยีคลาวด์คอมพิวเตอร์ เพื่อส่งเสริมทักษะวิถีแห่งการคิดเศรษฐกิจพอเพียงของนักเรียนชั้นมัธยมศึกษาปีที่ 6 ในภาคเรียนที่ 2/2563 พบว่ากลุ่มทดลองมีความพึงพอใจต่อกิจกรรมการเรียนรู้ในระดับมากที่สุด

4. สรุป

สถาปัตยกรรมเครื่องมือการพัฒนาแอปพลิเคชันขั้นสูงแบบคลาวด์ ด้วยแพลตฟอร์มคอนเทนเนอร์ ได้ใช้โปรแกรมภาษาพีเอชพีในการสร้างส่วนลงชื่อเข้าใช้ของผู้ใช้งานระบบ เมื่อลงชื่อเข้าระบบ ระบบจะสร้างคอนเทนเนอร์สำหรับผู้ใช้งานหรือผู้เรียนด้วย Docker ให้ผู้ใช้งานได้คอนเทนเนอร์สำหรับการพัฒนาแอปพลิเคชันขั้นสูง การทำงานภายในสถาปัตยกรรมเครื่องมือประกอบด้วย 3 ส่วน ส่วนเนื้อหาบทเรียนแบบปฏิบัติการ ส่วนที่สองเป็นส่วนเครื่องมือในการพัฒนาแอปพลิเคชันขั้นสูงที่เปิดเทอร์มินัลให้ใช้ในการทำงานรันผลลัพธ์การพัฒนาแอปพลิเคชันขั้นสูงและแสดงผลลัพธ์ของแอปพลิเคชันขั้นสูงในส่วนที่สามพร้อมทั้งสามารถจัดการฐานข้อมูลในส่วนนี้ด้วย ผู้ดูแลระบบสามารถจัดการคอนเทนเนอร์ทั้งหมดโดยผ่าน Portainer เมื่อพัฒนาเสร็จแล้วนำไปให้ผู้เชี่ยวชาญทดสอบผลความคิดเห็นของผู้เชี่ยวชาญอยู่ในระดับมากมีค่าเฉลี่ย 4.45 ส่วนเบี่ยงเบนมาตรฐาน 0.458 และเมื่อนำไปทดลองกับผู้ใช้งานหรือผู้เรียนจำนวน 77 คน ที่ได้เข้ามามีการศึกษาในภาคเรียนที่ 2-3 ปีการศึกษา 2565 พบว่ามีความพึงพอใจระดับมากมีค่าเฉลี่ยรวม 4.37 ส่วนเบี่ยงเบนมาตรฐาน 0.722 ดังนั้น ผู้วิจัยจึงเห็นว่าสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นได้ปรับใช้เทคโนโลยีด้วยดอกเกอร์ ซึ่งเป็นเครื่องมือสำหรับการสร้างและใช้งานคอนเทนเนอร์ ที่มีการจัดเตรียมเฟรมเวิร์กที่สามารถแยกบริการต่าง ๆ ได้ โดยแต่ละบริการมีฟังก์ชันการทำงานเป็นของตัวเอง แต่ในขณะเดียวกันก็มีการประสานงานอย่างเหมาะสม การนำสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นมาช่วยในการจัดการเรียนการสอนการพัฒนาแอปพลิเคชันขั้นสูงที่จำเป็นต้องเตรียมเครื่องมือในการพัฒนาจำนวนมากทำให้เกิดความยุ่งยากในการจัดเตรียมห้องปฏิบัติการขั้นสูงเพื่อรองรับผู้เรียนจำนวนมาก สถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นจึงทำให้ผู้เรียนสะดวก และผู้เรียนได้เข้ามามีศึกษาเนื้อหาวิชาได้ตลอดเวลาจะช่วยเพิ่มประสิทธิภาพในการเรียนมากยิ่งขึ้น ข้อเสนอแนะเพิ่มเติมคือในส่วนของจัดการการเรียนการสอนนั้น เมื่อผู้เรียนเกิดปัญหาขึ้นขณะที่ใช้งานสถาปัตยกรรมเครื่องมือที่พัฒนาขึ้นยังไม่มีระบบบริหารจัดการขอความช่วยเหลือจากผู้สอนเมื่อผู้เรียนเกิดปัญหาพร้อม ๆ กัน และยังไม่มียุทธวิธีที่ใช้ในการประเมินผลการเรียนของผู้เรียน

5. กิตติกรรมประกาศ

คณะผู้วิจัยขอขอบคุณภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏ

เชียงใหม่ที่ให้การสนับสนุนและอำนวยความสะดวกด้านต่าง ๆ สำหรับในการทำวิจัยครั้งนี้

6. เอกสารอ้างอิง

- Amazon Inc. 2023. **Amazon Web Services (AWS)**. Available Source: amazon. <https://aws.amazon.com>, Jun 5, 2023.
- Bamroongcheep, U. 2020. Development of the Flipped-Classroom Model via Google Cloud Computing and Social Media to Enhance Way of Thinking Skills in 21st Century for Teacher Students. **Journal of MCU Peace Studies** 8(4): 1552-1566. (in Thai)
- Charoenphuthiwat, T. and Buranasaksee, U. 2023. A Model-Based Tool on Kubernetes Platform for Selecting the Most Cost Effective Resources for Web Applications t, pp. 540-547. **In The 15th NPRU National Academic Conference 2023 Research and Development Institute**. Nakhon Pathom Rajabhat University, Nakhon Pathom. (in Thai)
- Docker Inc. 2023. **Use containers to Build, Share and Run your applications on Docker's web site**. Available Source: <https://www.docker.com/resources/what-container/>, June 5, 2023.
- Gary, B.S. and Harry, J.R. 2011. **System Analysis and Design** (9th ed). Cengage Learning, USA.
- Google Inc. 2023. **Google Cloud Platform**. Available Source: <https://cloud.google.com>, Jun 5, 2023.
- Gitpod GmbH. 2023. **GitLab**. Available Source: <https://www.gitpod.io/>, Jun 5, 2023.
- Laisema, S. 2018. Mobile Cloud Learning. **Journal of Education Silpakorn University** 16(1): 52-64. (in Thai)
- Mell, P. and Grance, T. 2011. The NIST definition of cloud computing. **National Institute of Standards and Technology Special Publication (NIST SP)** 53: 1-7.
- Microsoft Inc. 2023. **Azure**. Available Source: <https://azure.microsoft.com/en-us/>, Jun 5, 2023.
- Srisa-ard, B. 2017. **Basic Research**. Suweeriyasan, Chulalongkorn University, Bangkok. (in Thai)

- Thitipetchakul, C., Sompong, N. and Rampai, N. 2020. Development of Learning Management Model on Cloud Computing System Based on Connectivism to Enhance Information and Communication Technology Literacy for Undergraduate Students. **Ratchphruek Journal** 18(1): 38-48. (in Thai)
- Wanichbancha, K. 2011. **Statistical Analysis: Statistics for Decision Making**. Chulalongkorn University, Bangkok. (in Thai)
- Wannasin, W., Bamroongcheep, U. and Manosuttirit, A. 2021. The effects of active learning activities by using problem based Learning Via Cloud Computing Technology to Enhance Ways of Sufficiency Economy Thinking Skills of Twelfth Grade Students. **Srinakharinwirot Academic Journal of Education** 22(2): 132-151. (in Thai)