

Term Project Report: FurEveryHome

กลุ่ม Riyal or Fakay?

รายชื่อสมาชิก

นายพนพล	จุ่นเจิม	6509650146
นางสาวขวัญดาว	ขันบุรี	6509650245
นางสาวปุณณภา	ตรีรัตน์ดิลกกุล	6509650559
นายธีรภัทร	เกิดไพบูลย์	6509650468

เสนอ

ผศ.ดร.ประภาพร รัตนธำรง

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา
คพ.360 หัวข้อเลือกสรรด้านวิศวกรรมซอฟต์แวร์ ภาคเรียนที่ 1 ปีการศึกษา 2567
คณะวิทยาศาสตร์และเทคโนโลยี สาขาวิทยาการคอมพิวเตอร์
มหาวิทยาลัยธรรมศาสตร์

2. ข้อมูล Repository

Project Repository:

https://github.com/Kwandao6509650245/CS360 Project.git

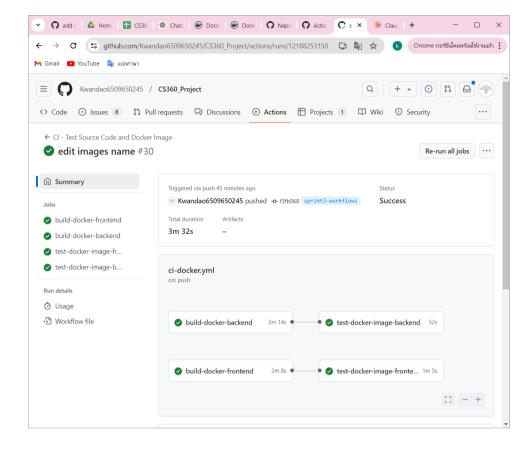
- 3. รายงานผลเกี่ยวกับเวลาที่ใช้ตั้งแต่ push code จนนำแอปพลิเคชันขึ้นให้บริการบน EC2
- a. แบบ Manual (การทดสอบโค้ด การติดตั้ง Server)
 - เวลา Integrate/Build Code: 9.43 นาที
 - เวลา Deploy Code ขึ้น Server: 11.04 นาที
 - รวมเวลาทั้งหมด: 20.47 นาที

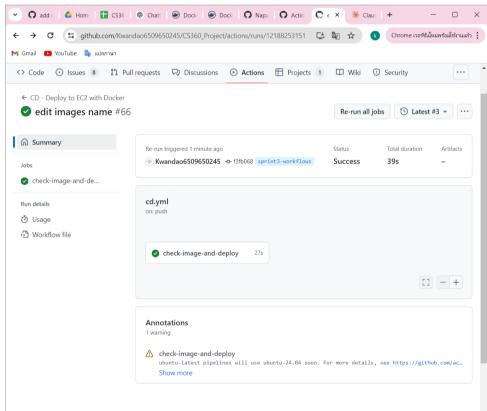
การทดสอบและติดตั้งเชิร์ฟเวอร์แบบ Manual ต้องทำขั้นตอนทั้งหมดด้วยตนเอง ตั้งแต่การทดสอบโค้ดไปจนถึงการ ตั้งค่าระบบเซิร์ฟเวอร์ ซึ่งอาจใช้เวลาเนื่องจากต้องตั้งค่าและแก้ปัญหาที่เกิดขึ้นระหว่างการติดตั้ง เสี่ยงต่อการเกิด human error

- b. แบบ Semi-Automate (มีการใช้ Bash script ช่วยการติดตั้ง Server)
 - เวลา Integrate/Build Code: 5.17 นาที
 - เวลา Deploy Code ขึ้น Server: 1.11 นาที
 - รวมเวลาทั้งหมด: 6.28 นาที

การใช้ Bash script ช่วยในการติดตั้งเซิร์ฟเวอร์ทำให้กระบวนการติดตั้ง dependencies เป็นไปอย่างสะดวก รวดเร็วขึ้น โดยไม่ต้องตั้งค่าติดตั้งทีละขั้นตอน แต่ยังคงต้องทำการตรวจสอบ

- c. แบบ Automate (ดำเนินการด้วย CI/CD Pipeline ที่สร้างขึ้น)
 - เวลา Integrate/Build Code: 3.32 นาที
 - เวลา Deploy Code ขึ้น Server: 0.39 นาที
 - **รวมเวลาทั้งหมด**: 4.11 นาที





• การใช้ CI/CD Pipeline ทำให้กระบวนการทั้งการ Build และ Deploy เป็นไปโดยอัตโนมัติ ลดเวลาในการทำงาน และข้อผิดพลาดที่อาจเกิดจากขั้นตอนด้วยมือ และมีการ automate test อยู่เสมอ

4. อภิปรายเกี่ยวกับประโยชน์ของกระบวนการ CI/CD

กระบวนการ CI/CD (Continuous Integration / Continuous Deployment) ช่วยลดเวลาในการพัฒนาและติดตั้ง แอปพลิเคชัน โดยสามารถทำให้โค้ดที่เขียนเสร็จแล้วสามารถนำไปทดสอบและติดตั้งขึ้นเชิร์ฟเวอร์ได้โดยอัตโนมัติ กระบวนการ นี้มีข้อดีหลายประการ ดังนี้

- ลดเวลาในการ Deploy: จากที่ต้องทำการติดตั้งเชิร์ฟเวอร์ด้วยมือหลายขั้นตอน ในกรณีที่ใช้ CI/CD Pipeline เวลา ที่ใช้ในการ Deploy ลดลงอย่างมาก และเป็นการติดตั้งแบบ automate เมื่อมีการอัพเดตซอฟแวร์จึงเป็นไปได้อย่าง สะดวกและรวดเร็วขึ้น
- ลดข้อผิดพลาด: การใช้ Pipeline ช่วยให้กระบวนการต่าง ๆ มีความสม่ำเสมอและสามารถลดข้อผิดพลาดที่เกิดจาก human error
- ความถูกต้องของโค้ด: กระบวนการ CI/CD สามารถทำการทดสอบโค้ดทุกครั้งที่มีการเปลี่ยนแปลง ทำให้มั่นใจได้ว่า โค้ดที่ได้รับการ Deploy เป็นโค้ดที่ผ่านการทดสอบแล้ว

จากการพัฒนาโปรเจคนี้ การสร้าง CI/CD Pipeline มีประโยชน์มาก เนื่องจากช่วยประหยัดเวลาในระยะยาว ทำให้ สามารถโฟกัสกับการพัฒนาและปรับปรุงแอปพลิเคชันได้มากขึ้น ดังนั้น จึงควรทำ CI/CD ในโครงการนี้ เพราะจะทำให้ กระบวนการ Deploy มีประสิทธิภาพสูงสุด ลดข้อผิดพลาดที่เกิดจากการกระทำด้วยมือ ช่วยลดระยะเวลาในขั้นตอนต่างๆ และสามารถจัดการกับการพัฒนาแอปพลิเคชันในระยะยาวได้ดียิ่งขึ้น